IMAGE CORPUS REPRESENTATIVE SUMMARIZATION

Anurag Singh, Lakshay Virmani*, and A.V. Subramanyam, Member, IEEE*

Division of Information Technology, Netaji Subhash University of Technology*, Delhi, India
Indraprastha Institute of Information Technology, Delhi, India

ABSTRACT

We propose a novel approach for image corpus representative summarization using GAN. Our technique can be used to automatically provide a condensed set of representatives for the given image collection. The generated summary can be used for rapid prototyping as models can be trained using the summarized set instead of the larger original dataset. The problem is challenging because a good summary must cover various aspects of an image set such as relevance and diversity. Additionally, lack of sufficient ground truth data makes the problem hard to solve using classical supervised machine learning approaches. In our algorithm, we use CNN and an MLP score layer to compute the priority of each image towards the summary. Our network is trained in an unsupervised manner using a generator for reconstructing the original dataset, and a discriminator, for classifying between original and summary. We show the efficacy of the algorithm using rigorous experiments.

***Index Terms*—** Summarization, GAN, unsupervised learning.

1 Introduction

Image corpus representative selection or summarization is an essential requirement for efficient representation, navigation and exploration [1]. Web image collection for e-commerce, tourism and travel exploration, story-telling from personal album collections, online image recommendation systems are some of the immediate applications of automatic image corpus summarization [2]. Another very important application is while training machine learning algorithms. In particular, deep networks require significant amount of time even for fine-tuning. In such a case, it would be beneficial if the network can be fine-tuned using a representative summary instead of original dataset. However, an annotated summary is not readily available. Thus, a summary which can be labeled at a much less cost compared to original dataset would be beneficial. *Summarization of dataset can help train models without trading-off much on accuracy as the diversity of data is maintained while saving huge computational resources*. Once the appropriate hyper-parameters are known, the model can be appropriately scaled.

Summarization has been well explored in videos for efficient browsing and other applications [3], [4], [5], though image corpus summarization has not received an equivalent attention [6, 7, 8, 9]. This is partly because, videos have the redundancy in the temporal dimension which can be effectively learnt and reduced, whereas image corpus need not have such redundancy. Thus the problem becomes quite challenging. However, there is a lot of redundancy in the intra class samples which can be exploited to determine the most representative samples. Summary of image corpus can be both qualitatively and quantitatively analyzed based on factors of relevance and diversity, which we define as following.

- *Relevance* means how relevant is a particular image to a given task such as classification, segmentation or detection. For example, a certain image may be relevant for a specific task and should be captured in the summary, whereas, for a different task this image may not be desired as a part of the summary.
- *Diversity* maintains that all the images that are distinct are included in the summary and must not contain any redundancy [10], [11].

Towards this, we make the following contributions.

- We propose a novel image corpus summarization model using MLP and GAN. Our network can be trained in an end-to-end fashion.
- We introduce a task-specific loss to generate the summary related to a given task. In our experiments related to classification task, we show that the loss promotes both relevant and diverse images in summary.
- We analyze the relevance and diversity of summary using multiple metrics. We also analyze the goodness of summary by training a classifier on it and compare the performance against usage of original dataset. In our experiments, we demonstrate that the accuracy achieved by fine-tuning a model using the summary is comparable to the accuracy achieved when original dataset is used.

2 Related Works

Image collection summarization can be categorised into different class of works namely, summary based on sparsity learning and personal albums.

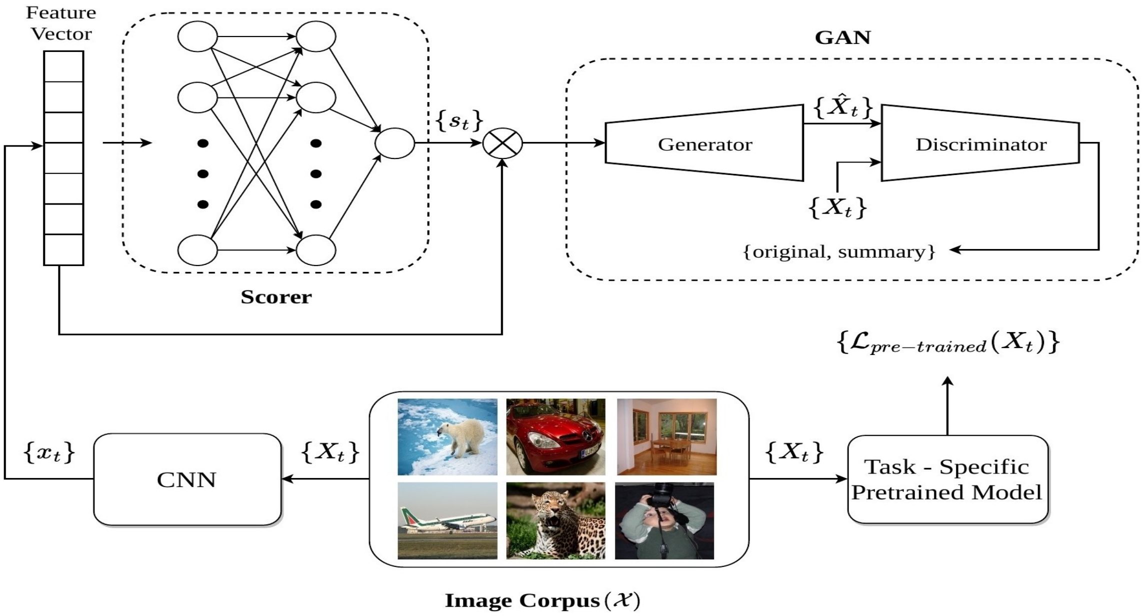


Fig. 1: Images X_t are given as an input to CNN. The scorer ANN $\mathcal{S}$ gives score s for each image which defines its importance for the summary. Then each feature vector is weighted by its score and forwarded to generator called as $\mathcal{G}$ for reconstructing the image $\hat{X}$. The discriminator $\mathcal{D}$ classifies $\hat{X}$ as original or summary class. The pre-trained classifier model is used when a classification task specific summary needs to be generated. For generating the summary, we can select the images based on the score s.

Sparsity Learning: In [2], Yang *et al.* formulate image summarization as an optimization problem. The authors apply a dictionary learning approach based on SIFT-Bag of Words model for creating the summary. On similar lines, [12] present a structured sparsity learning approach for determining representative samples. The authors make use of three different regularizers - group sparsity, diversity and locality-sensitivity to achieve the task. Other works based on dictionary learning approach are [13], [14]. In [7], Wang and Yuan propose to learn the representative samples by minimizing the L2 distance between the selected samples and the center obtained by the samples in the kernel space. An L1 constraint is applied on selection vector to promote sparsity in the representative examples.

Personal Albums: Some works generate summary from a large personal photo albums [11], [15]. In [11], Sinha et. al. propose to use multdimensional features such as visual, temporal, event type, location and people. In [16], authors extract storyline from the album photos.

Summary can be generic or task specific. Algorithms presented in [2], [13], [14], [12] provide generic summary as they focus on reconstruction of complete dataset using the summary itself. Whereas, [11], [15] are task specific summaries where the task is storytelling. *In this paper, we propose a model which can provide both generic as well as task specific summary. Our work is very different from the aforementioned algorithms as it employs a multilayer perceptron scorer and a generative adversarial network to generate the summary. To the best of our knowledge, this is the first deep learning based approach which can jointly perform generic and task-specific image corpus summarization. The major advantage of our work is that the network is trained in an unsupervised end-to-end manner and does not suffer from inherent issues of handcrafted features which are used in dictionary learning approaches*. In the following, we describe our architecture.

3 Image Collection Summarization

Our network takes CNN feature embedding of images as an input. CNN is followed by a scorer which is a multilayer perceptron. The scorer assigns a relative importance score to each image such that the higher the score the more the likelihood of the image being present in summary. The fused output of score layer with CNN feature vectors are used as an input to GAN. The function of the generator is to reconstruct the images of original dataset. The discriminator then distinguishes between the original and the reconstructed dataset. A detailed diagram is given in Figure 1.

3.1 Problem Formulation

Given a collection of n images $\mathcal{X} = \{X_1, X_2, ..., X_n\}$, we aim to find a subset summary of these images while preserving the relevance and diversity.

3.2 Learning Framework

In our algorithm, we first extract features of the images $\mathcal{X} = \{X_1, X_2, ..., X_n\}$ using GoogLeNet [17]. Let these features be $x = \{x_t : t = 1...n\}$. These features are fed as an input to the scorer ANN $\mathcal{S}$ which consists of 2 layers of 1024 neurons each, followed by a single neuron that outputs the score corresponding to an image. These scores represent the relative importance of the image being present in the summary. Images with higher scores can be selected to generate the summary. Let $s = \{s_t\},\ s_t \in [0,1]$ be the scores. The features x_t are weighted using these scores. This then acts as an input to the generator $\mathcal{G}$ which reconstructs the image collection denoted by $\hat{\mathcal{X}} = \{\hat{X}_1, \hat{X}_2, ..., \hat{X}_n\}$. The discriminator $\mathcal{D}$ in the GAN is aimed to classify images into two distinct classes, and hence, distinguish between images from $\mathcal{X}$ and $\hat{\mathcal{X}}$ as Original and Summary. Generator and discriminator are trained adversarially until the discriminator is not able to discriminate between the summary and original dataset.

3.3 Training the model

We discuss the different loss functions and training part of the algorithm in this section. The parameters of the model are w_s, w_g and w_d for the scorer, generator, and discriminator, respectively. The training of our model is defined by following losses: reconstruction loss $\mathcal{L}_{reconstruct}$, loss of GAN $\mathcal{L}_{GAN}$, regularization loss $\mathcal{L}_{sparsity}$ and task-specific loss $\mathcal{L}_{task-specific}$. We learn w_s by minimizing $\mathcal{L}_{sparsity} + \mathcal{L}_{reconstruct} + \mathcal{L}_{task-specific}$, w_g by minimizing $\mathcal{L}_{GAN} + \mathcal{L}_{reconstruct}$ and w_d by maximizing $\mathcal{L}_{GAN}$.

$\mathcal{L}_{task-specific}$ is added only for the case of task-specific summarization. We explain this in more detail under subsection 3.3.4. The training pipeline is also illustrated in Algorithm 1. We train the model using a combination of losses and we name each variant in Section 5.

Algorithm 1: Training the model

```
function UPDATE PARAMS   ▷ where input is the feature vector sequence and output is learned parameters $w_s, w_g, w_d$
    for max number of iterations do
        $X \leftarrow$ mini batch of images
        $x \leftarrow \mathtt{CNN}(X)$
        $s \leftarrow \mathcal{S}(x)$          ▷ compute scores
        $E \leftarrow sx$          ▷ weigh feature vectors by scores
        $\hat{X} \leftarrow \mathcal{G}(E)$          ▷ Reconstruction
        % Update rules:
        $w_s \overset{+}{\leftarrow} -\nabla(\mathcal{L}_{reconstruct}+\mathcal{L}_{sparsity}+\mathcal{L}_{task-specific})$
        $w_g \overset{+}{\leftarrow} -\nabla(\mathcal{L}_{reconstruct}+\mathcal{L}_{GAN})$
        $w_d \overset{+}{\leftarrow} \nabla(\mathcal{L}_{GAN})$
```

3.3.1 Reconstruction Loss $\mathcal{L}_{reconstruct}$

This loss is used to make a summary that captures all diverse images. If original set of images can be reconstructed using the feature vectors of the summary, then summary can be considered to have contained all diverse images [2]. The reconstruction error is given as,

$$\sum_{t=1}^{n} \left\| X_t - \hat{X}_t \right\|_2 \tag{1}$$

where X_t is an image in the original dataset, $\hat{X}_t$ is the corresponding reconstructed image from the generator and n is the number of images in original dataset. We try to achieve a similar formulation where we take feature vectors from our summary and try to reconstruct the image dataset in feature space. Thus we express each feature vector x_t as a linear combination of the summary image's feature vectors as,

$$x_t = \sum_{j=1}^{\sigma n} d_j v_{jt} \tag{2}$$

where d is the feature vector which corresponds to images that are part of the summary, v_{jt} is the non-negative weight of vector d_j, and σ is a hyper-parameter to control the fraction of images selected for the summary. Then, we define the reconstruction error as,

$$\mathcal{L}_{reconstruct} = \sum_{t=1}^{n} \left\| x_t - \sum_{j=1}^{\sigma n} d_j v_{jt} \right\|_2 \tag{3}$$

We compute the weights v_{jt} using multiplicative algorithm in [18].

3.3.2 Loss of GAN $\mathcal{L}_{GAN}$

The reconstruction loss can only reconstruct the feature vectors using representatives. However, apart from the reconstruction error there is no good measure to determine whether the representatives summarize the dataset well. Therefore, we use an unsupervised way of determining the goodness of summary using GAN. The GAN is trained to discriminate whether the generated images belongs to original set or summary. Towards this, the input to GAN is the weighted feature vector $s_t x_t$. We use DCGAN in our model [19]. The $\mathcal{L}_{GAN}$ is thus defined as,

$$\mathcal{L}_{GAN} = log(\mathcal{D}(X)) + log(1 - \mathcal{D}(\mathcal{G}(s_t x_t)))) \tag{4}$$

where $\mathcal{G}$ is the generator and $\mathcal{D}$ is the soft-max output of the discriminator. Initially, the discriminator can easily classify X as the original image and the generated images as summary. However, once the network is trained, the discriminator does not clearly distinguishes between the original and summary examples.

3.3.3 Regularization loss $\mathcal{L}_{sparsity}$

This loss regularizes the number of images that form the summary by learning the scores s_t which represent the relative importance of the feature x_t in the summary. Regularization is required to ensure that the summary length is minimal. We use length regularizer loss as well as DPP loss given by

$$\mathcal{L}_{sparsity} = \mathcal{L}_{LR} + \delta \mathcal{L}_{DPP} \tag{5}$$

where,

$$\mathcal{L}_{LR} = \left\| \frac{1}{n} \sum_{t=1}^{n} s_t - \sigma \right\|_2$$

$$\mathcal{L}_{DPP} = -log(P(s_S))$$

where $\delta \in \{0,1\}$ and $\mathcal{L}_{DPP}$ is the Determinantal Point Process (DPP) loss [20]. DPP loss has also been popularly used for summarizing videos [21], [3]. DPP loss promotes diversity while minimizing the number of images in the summary. In our case, we construct the DPP loss as follows. For set of images $\mathcal{X}$, we select a subset $S \subset \mathcal{X}$ which constitute the images in summary with corresponding scores s_S. We first compute a distance matrix $D \in \mathcal{R}^{n \times n}$. D is constructed with each entry as $D_{i,j} = s_i s_j x_i^T x_j$. We then compute the probability of the scores s_S assigned by DPP, and we have,

$$P(s_S; D) = \frac{|D(s_S)|}{|D+I|} \tag{6}$$

where $|.|$ denotes determinant, I is an identity matrix, and $D(s_S) \in \mathcal{R}^{\sigma n \times \sigma n}$ is a submatrix of D given s_S. The value of σ is chosen according to the required size of the summary. Thus, by varying σ we control sparsity in terms of summary length. For low values of σ, say $\sigma = 0.01$ (1% summary size), the summary will be much more sparser as compared to when $\sigma = 0.1$ (10% summary size). In other words, the size of the summary would be σn.

3.3.4 Task-specific loss $\mathcal{L}_{task-specific}$

A task specific summary would be used to perform certain task and must perform best for that intended task. In the following, we assume a task specific summary where the task is classification. To this end, we introduce a task-specific loss,

$$\mathcal{L}_{task-specific} = \frac{(1-s)\mathcal{L}_{pre-trained}(X)}{\beta} \tag{7}$$

where $\mathcal{L}_{pre-trained}(X)$ is the loss obtained from the task specific pre-trained model and β is a hyper-parameter. β also controls the number of outliers to be selected in summary from the dataset. In our experiments, $\mathcal{L}_{pre-trained}$ is the cross entropy loss. We explain the role of β in Section 5. The task-specific loss promotes the presence of outliers, where outliers represent both relevance and diversity. We illustrate this using the following scenario.

Let us consider a case where there is a dense cluster of points and an isolated point. If we want to generate a summary, we can sample an arbitrary point from a cluster as it represents the entire set of points in the cluster. However, we miss on the diversity aspect as the isolated point is non-redundant and should be present in the summary. Further, for these outliers, the classification loss is high as they are less probable candidates for correct classification. Furthermore, if these outliers have a low score s, they are less likely to be present in the summary. Now, during the process of minimization of this task-specific loss to increase classification accuracy, the score s must increase, thereby increasing the likelihood of those images being present in the summary. *Thus by encouraging these outliers, the training of the network can also be robust to these set of exemplars, thereby selecting the relevant samples which can enhance the task-specific performance.* In case of general summary, the task-specific loss is not incorporated in the overall objective function.

4 Datasets

We use publicly available datasets to train and test our model. Since there are very few datasets which have summary annotations in terms of relevance and diversity, we test on both datasets - where annotations are available as well as where it is unavailable. We evaluate our model using following datasets: CIFAR-10 [22], CIFAR-100 [22], Animals with Attributes 2 (AwA2) [23], VOC2012 [24] and Diversity 2016 [25]. CIFAR-10 consists of 60,000 32×32 tiny images belonging to 10 classes with similar images per class. There are 50,000 training and 10,000 test images. CIFAR-100 consists of 60,000 32×32 tiny images with 600 images per class. The classes are divided into 20 super-classes with 5 classes per super-class. AwA2 is another data set used for classification purpose. It contains 37,322 images of 50 animal classes. Visual Object Classes (VOC2012) is another image classification data set with 20 classes and 11,530 images. Diversity 2016 contains the images with corresponding ground truth images for task of diversity in image retrieval. Images are ranked according to their importance within a class. We use Top-50 ranked images from each class to create the ground truth. There are 20,821 images of multiple classes with each class containing approximately 300 images.

5 Experiments

We implement our architecture using TensorFlow[1]. All the experiments are performed on Nvidia GTX 1080 GPU. We train the network with a batch size of 32 images for 25 epochs. We use the output of Pool5 layer of the GoogLeNet [17] of 1024-dimensions for the feature descriptor. The values of the hyper-parameters are empirically set. A learning rate of 0.0001 and 0.0002 is used in the Adam optimizer for the scorer and the GAN, respectively. Batch normalization and dropout layers are also employed. We use a three-layer ANN, with 1024 hidden units each, in the first two layers, followed by a single unit as the output layer. For the task-specific loss ($\mathcal{L}_{pre-trained}$), Inception V3 model is fine-tuned for the images in the corpus, and the cross-entropy loss is computed.

The different metrics that are used to evaluate summary are reconstruction error, F-score, Gini index [26], and task accuracy. We use multiple losses and denote the model using specific losses with the following names. When we use $\mathcal{L}_{reconstruct} + \mathcal{L}_{GAN} + \mathcal{L}_{LR}$, we call this variant as SUM_{gen} and it gives a general summary as the model does not have any notion of a pre-defined task. We denote the summary with loss $\mathcal{L}_{reconstruct} + \mathcal{L}_{LR} + \mathcal{L}_{DPP} + \mathcal{L}_{GAN}$ as SUM_{gen}^{DPP} which also gives a general summary. When we use $\mathcal{L}_{reconstruct} + \mathcal{L}_{LR} + \mathcal{L}_{GAN} + \mathcal{L}_{task-specific}$, we denote it as SUM_{task} and it generates a task-specific summary.

[1] Our code is available at https://github.com/anonblindreview/Summarization

5.1 Reconstruction Error

We report the reconstruction error in Table 1. We observe that the reconstruction error gets better with increase in the fraction of images retained in summary. This is expected as with increase in σ, the summary size increases and it becomes easier to reconstruct the original dataset. Further, compared to generic summary obtained using SUM_{gen} or SUM_{gen}^{DPP}, it can be seen that the reconstruction error is lower when the task-specific loss is incorporated. This is because minimizing the task-specific loss ensures diversity by prioritizing the inclusion of the outliers, as discussed in Section 3.3.4. A balanced presence of the outliers in the summary helps in the reconstruction of all the different kind of images present in the original dataset. This presence of outliers is further controlled by the value of β. This is empirically shown by evaluating the number of outliers and the reconstruction error for different values of β, for the VOC2012 dataset with σ = 0.03. In Fig. 2-Left, we find that the number of outliers decreases with increase in the β. Further, in Fig. 2-Right, we observe that when β is low, the reconstruction error is high, as the summary in that case contains a significant amount of outliers, and hence the difficulty in reconstructing the original dataset where inliers are in large quantity. The optimal value of β is achieved when there is a balanced presence of outliers and at this value of β the reconstruction error is minimum.

Table 1: Reconstruction Error for different values of σ

σ	Method	CIFAR10	CIFAR100	AWA2	VOC2012
0.01	SUM_{gen}	0.307	0.314	0.515	0.565
	SUM_{gen}^{DPP}	0.292	0.283	0.509	0.546
	SUM_{task}	0.274	0.268	0.508	0.542
0.03	SUM_{gen}	0.286	0.272	0.412	0.539
	SUM_{gen}^{DPP}	0.252	0.248	0.416	0.469
	SUM_{task}	0.251	0.257	0.410	0.459

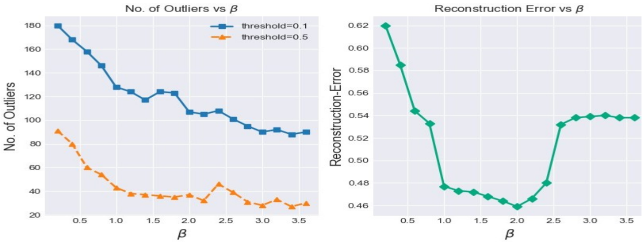


Fig. 2: Figure highlighting variation of outliers and reconstruction error vs. β. The thresholds used here are 0.1 and 0.5, all images with cross-entropy loss ($\mathcal{L}_{pre-trained}$) greater than the threshold are considered to be outliers

5.1.1 F-Score

We report precision, recall and F-score *vs* σ in Table 2. Precision is the ratio of the number of overlapping images between summary and ground truth to that of summary length. Recall is the ratio of the number of overlapping images between summary and ground truth to that of ground truth size. The F-score is harmonic mean of the two. We observe a high recall and F-score when the summary length is 50% of original dataset. In this case, we observe a low precision compared to recall as the summary length is approximately 10,000 against 3500 for ground truth. We find that recall and F-score are very low for $\sigma = 0.01,\ 0.03$. In these cases, the summary length is about 200 and 600 respectively, which is very low compared to ground truth size. However, it may be necessary to generate such small summaries in order to save computational resources. We observe a best precision of 19.6% for $\sigma = 0.03$ with SUM_{gen} variant. Recall is highest for SUM_{task} model at $\sigma = 0.5$, whereas, best F-score of 25.67% is obtained with SUM_{gen}^{DPP} model at $\sigma = 0.5$.

Table 2: F-Score vs σ plot for Diversity 2016. Pr - Precision, Re - Recall, Fs - F-Scores (in %)

Method	σ	0.01	0.03	0.05	0.1	0.3	0.5
SUM_{gen}	Pr	16	**19.6**	19.31	16.26	17.77	16.94
	Re	2.05	3.08	4.80	6.857	29.28	47.60
	Fs	3.645	5.3446	7.68	9.64	22.12	24.99
SUM_{gen}^{DPP}	Pr	17.605	18.12	18.26	17.877	17.11	17.13
	Re	2.85	3.428	6.258	9.77	28.88	51.17
	Fs	4.916	5.766	9.321	12.63	21.49	**25.67**
SUM_{task}	Pr	16.91	16.20	18.16	15.02	16.62	17.11
	Re	2.94	3.20	4.857	8.371	28.94	**51.25**
	Fs	5.01	5.3447	7.66	10.75	21.11	25.65
k-means	Pr	11.53	14.42	15.85	15.75	17.62	17.06
	Re	0.68	2.57	4.71	9.37	30.62	47.05
	Fs	1.29	4.36	7.26	11.75	21.99	25.04

5.2 Gini Index

In order to measure diversity, we compute Gini index [26]. We compare these indices computed for randomly picked images from dataset, images picked corresponding to centers of k-means feature clusters and for images from summary, and plot in Fig. 3. In order to perform a comparison, we require that the summary generated using the k-means method is also of the same size. Towards this, the number of clusters in the k-means method is set to be equal to σn. *We use Gini index because this metric inherently gives uniform weightage to all the classes present in dataset. This is important as summary must comprise images from all classes.* In case of k-means and random summary generation, this uniformity may not be achieved. We observe that for generic summary of datasets CIFAR10 and CIFAR100, the Gini index is best (lowest) for SUM_{gen}^{DPP} which essentially means that the probability of choosing images from each class towards summary generation is more uniform than k-means and random, thus maintaining the diversity.

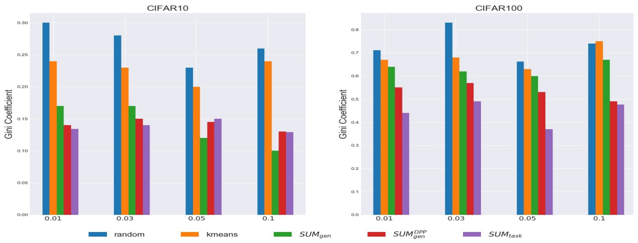


Fig. 3: Gini index for different datasets and σ values. X-axis represents σ and Y-axis represents Gini index. Proposed model gives best (lowest) Gini index compared to K-means and random methods

Table 3: Classification accuracies (in %) when original and summaries at 10%, 30% and 50% are used to fine-tune inception-v3

Dataset	Original	10%	30%	50%
CIFAR10	89.12	80.61	83.75	85.51
CIFAR100	65.35	51.85	58.71	61.07
AwA2	92.5	89.87	91.15	91.77
VOC2012	79.74	77.05	78.61	79.00

Table 4: Classification accuracy for various σ values. Best value for general summary among SUM_{gen}, SUM_{gen}^{DPP} and k-means are highlighted in bold

σ	Method	CIFAR10	CIFAR100	AWA2	VOC2012
0.01	SUM_{gen}	63.03	22.76	70.48	55.21
	SUM_{gen}^{DPP}	**65.77**	**25.71**	**71.93**	**57.35**
	SUM_{task}	68.70	31.51	72.95	58.56
-	k-means	64.97	24.86	71.06	55.25
0.03	SUM_{gen}	66.73	27.13	76.41	60.37
	SUM_{gen}^{DPP}	**71.09**	**32.49**	**76.22**	**62.90**
	SUM_{task}	72.06	33.51	77.84	64.08
-	k-means	67.62	32.30	74.91	60.75
0.05	SUM_{gen}	**72.67**	37.39	78.72	64.73
	SUM_{gen}^{DPP}	72.07	**37.44**	**78.78**	65.01
	SUM_{task}	78.08	41.00	80.56	66.07
-	k-means	71.03	34.92	77.08	**65.71**
0.1	SUM_{gen}	75.45	40.71	81.17	67.88
	SUM_{gen}^{DPP}	**75.73**	**42.40**	**82.39**	68.25
	SUM_{task}	78.89	46.26	83.55	**70.87**
-	k-means	71.96	39.09	78.12	70.35

5.3 Qualitative Analysis

In order to understand the effect of DPP and Task-specific losses, we perform the t-SNE visualization experiment for VOC2012 dataset with $\sigma = 0.05$ for the different variants of our model. In Fig. 5, we show the plots for full original VOC2012 in Fig. 4a and the summary at 5% with SUM_{gen} in Fig. 4b, with SUM_{gen}^{DPP} in Fig. 4c and with SUM_{task} in Fig. 4d. In Fig. 4c and 4d, we mark a cluster with solid contour. As DPP generates a summary which is more sparse, we can observe it in the highlighted part of Fig. 4c. We see that the cluster is sparser compared to the same cluster highlighted in Fig. 4d. Further, it can be seen that when task-specific loss is incorporated in the model, outliers get included in the summary. This can be observed from dashed contours in Fig. 4d. In case of outliers which are a part of clusters of other classes, the task-specific loss can capture them well.

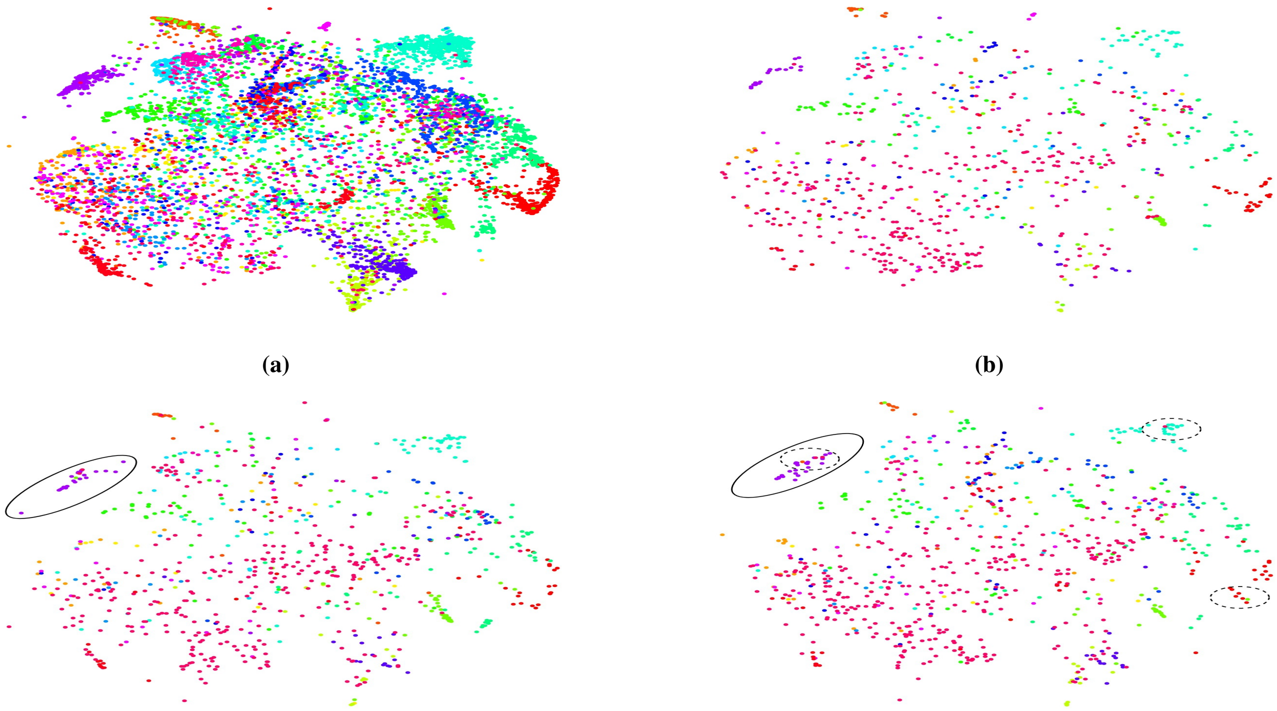
In Fig. 5a and 5b we show the t-SNE image embeddings of the ground truth summary and summary generated by our model. We mark the common images in a red boundary.

We illustrate the GAN generated images in Fig. 6 which are generated while training SUM_{gen} on CIFAR-10 dataset with $\sigma = 0.01$ and a batch size of 32 images. It can be seen that the generator is able to generate recognizable images even when trained for just 25 epochs.

5.4 Classification Accuracy

In case where class annotations are unavailable, we can create a general summary. Annotating a summary would be significantly less expensive compared to the original data. Further, if the general summary is a good representative of the original data, then the summary should give a performance close to what can be obtained using original data. In order to create a general summary, we use only $\mathcal{L}_{reconstruct}$, $\mathcal{L}_{GAN}$ and $\mathcal{L}_{sparsity}$ (with $\delta = 0$) losses. In order to evaluate whether the summary is a good representation of original, we perform the following experiment. We first fine-tune an inception v3 model on original dataset and compute the classification accuracy. Further, we fine-tune using summary only and compute the classification accuracy. We report these results in Table 3. Since our goal is to test the goodness of summary, we only run the model for few epochs attaining a decent accuracy, though state of art accuracy may be achieved by using more epochs. We observe that the accuracy achieved using summary is good when compared to its original counterpart. *These results are of great significance because the training using summary uses only 10% of original data, while the trade-off in accuracy is only about 6.8%.* In case of 30% and 50% summary of original data, the drop in accuracy is only about 3.62% and 2.34% respectively. Thus it is evident that the summary captures most of the aspects of the dataset.

We compare the accuracies of summary generated by Random sampling, k-means, SSDS [12], HyperSphere [7] and SUM_{gen}^{DPP}. In Table 5, for $\sigma = 0.1$, we observe that the proposed model performs better for CIFAR10 and CIFAR100. This is attributed to the fact that the scorer efficiently computes the relative importance of each image in summary. Also, the reconstruction error ensures that the feature representations of original dataset images can be recontructed using the feature vectors of summary images. Further, the GAN helps learn the summary images such that these images belong to the domain of original dataset itself.



(a)

(b)

(c)

(d)

Fig. 4: t-SNE plot for VOC2012. (4a) Full dataset, (4b) summary at 5% with SUM_{gen} , (4c) summary with SUM_{gen}^{DPP} and (4d) summary with SUM_{task} variants. Different colors represent different classes

Table 5: Classification accuracies (in %) comparison for summary at $\sigma = 0.1$

Method	CIFAR10	CIFAR100	AwA2
Original	89.12	65.35	92.50
Random	75.89	45.31	86.13
k-means	78.24	48.45	87.35
SSDS [12]	79.34	50.60	88.61
HyperSphere [7]	79.13	48.32	**88.90**
Ours	**80.61**	**51.85**	89.50

Evaluation under outliers

In some cases, it may be argued that the network trained with a summary may not perform well when the test set has outliers which are not captured in the summary. Thus, in order to determine the robustness of our model against outliers, we specifically inject them into the test sets. We report the accuracy without outliers in Table 3, and with outliers in Table 4. We experimentally define an outlier image which has $\mathcal{L}_{pre-trained} > 0.8$ for each dataset. The outlier injection is also carried out in order to observe the effectiveness of the task-specific loss. We can clearly see that the presence of task-specific loss significantly boosts the accuracy in all cases. As no such outlier injection is carried out in the experiments for Table 3, the reported accuracies are higher in its case.

Table 6: Time required for summary generation

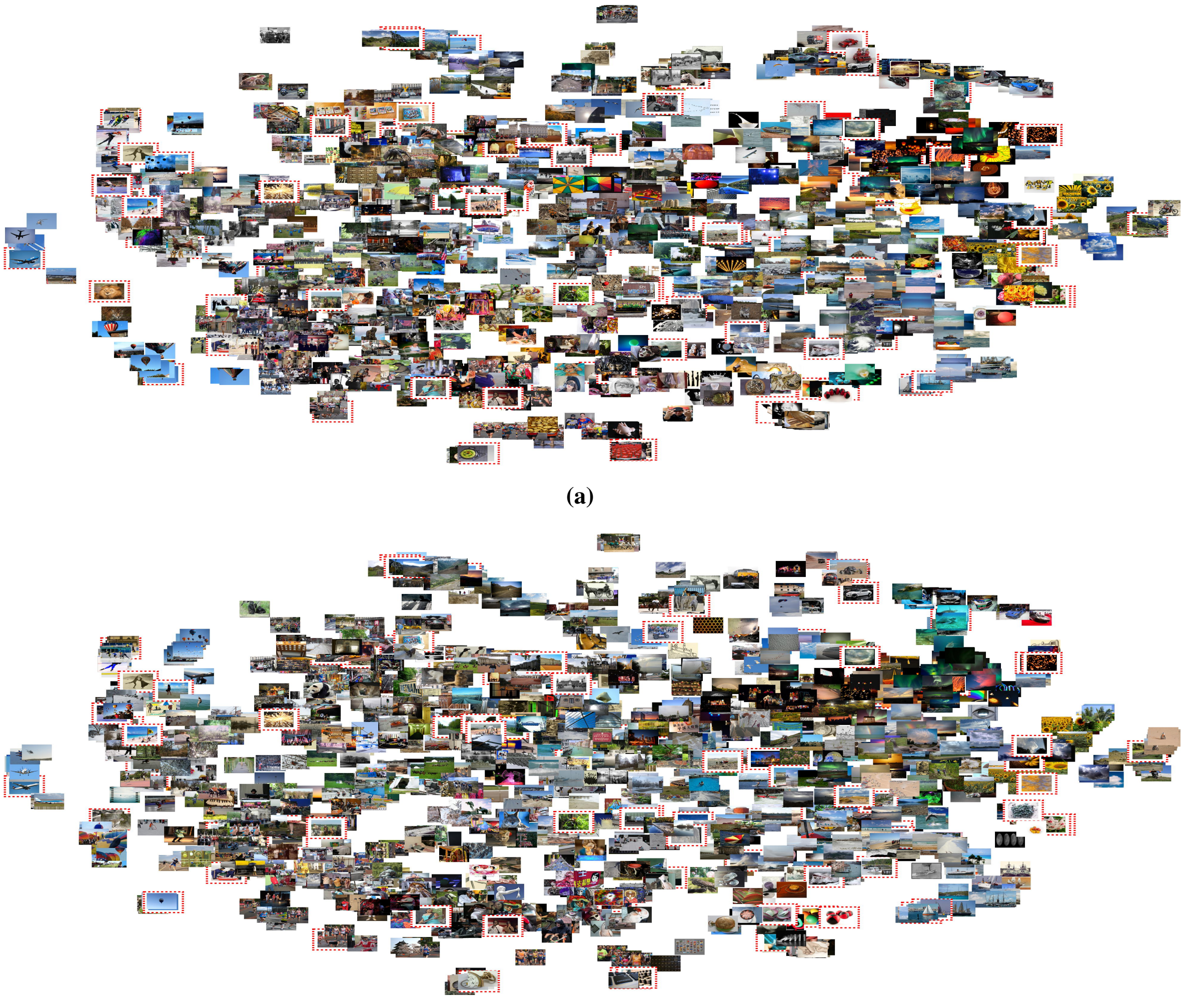
Dataset	No. of images	Resolution	Time
CIFAR-10	50000	32x32	3h
CIFAR-100	50000	32x32	3h
AwA2	32000	299x299	2h 15m
VOC2012	15000	299x299	1h
Diversity 2016	20821	299x299	1h 30m

5.5 Time Complexity

Here, we give the running time for each of the datasets. The training is run for 25 epochs with a batch size of 32 images. The average time required for training our model and generating the summary for the different datasets is given in Table 6. The datasets AwA2, VOC2012 and Diversity 2016 have images of varying resolutions, so for consistency, upsampling/downsampling is performed to get a constant resolution of 299x299 for all the images.

6 Conclusion

In this work, we propose an unsupervised model to create a summary out of a large collection of images. From the origi-



(a)

(b)

Fig. 5: (a) Ground truth summary for Diversity 2016, 1400 images (Top-20 images selected from each of 70 classes). Please zoom in for better visualization, (b) Summary generated by our model for Diversity 2016, 1400 images. The red boundaries highlight the images which are common to both summary and ground truth

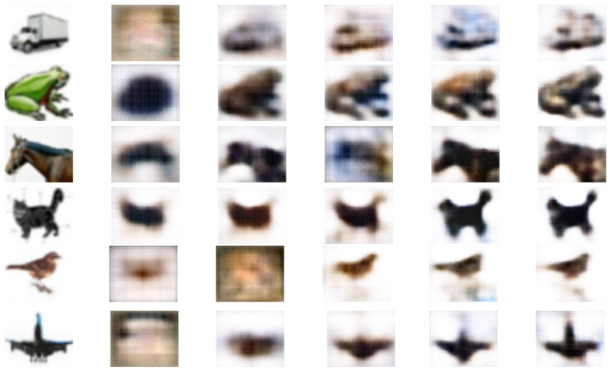


Fig. 6: Generated images for CIFAR-10. First column - original images; subsequent columns show images generated from epochs 5, 10, 15, 20 and 25 respectively

nal image set, our model selects the most diverse and relevant images. The proposed network makes use of a scoring layer and fusion of CNN feature vectors with scores as input to GAN. We train the model using four different losses namely reconstruction, GAN, sparsity and task-specific loss. We evaluate the algorithm with various metrics and show the efficacy of our network. In addition, we show that the classification results attained by training a deep network on summary only and on original dataset are comparable. Thus, the summary can be used for a quick analysis of various models without needing to train on entire dataset. In case the labels are not available, our technique can be used to summarize and retain a fraction of data, which can be relatively convenient to annotate. Further, one can perform different tasks on this data before scaling up the model or performing other processing on the original data.

7 References

[1] L. S. Kennedy and M. Naaman, "Generating diverse and representative image search results for landmarks," in *WWW*, 2008, pp. 297–306.

[2] C. Yang, J. Shen, J. Peng, and J. Fan, "Image collection summarization via dictionary learning for sparse representation," *Pattern Recognition*, vol. 46, no. 3, pp. 948–961, 2013.

[3] B. Mahasseni, M. Lam, and S. Todorovic, "Unsupervised video summarization with adversarial lstm networks," in *Proceedings of the IEEE CVPR*, 2017, pp. 2982–2991.

[4] H. Gu and V. Swaminathan, "From thumbnails to summaries-a single deep neural network to rule them all," in *ICME*, 2018, pp. 1–6.

[5] C. Huang and H. Wang, "Novel key-frames selection framework for comprehensive video summarization," *IEEE TCSVT*, 2019.

[6] S. Tschiatschek, R. K. Iyer, H. Wei, and J. A. Bilmes, "Learning mixtures of submodular functions for image collection summarization," in *NIPS*, 2014, pp. 1413–1421.

[7] H. Wang and J. Yuan, "Representative selection on a hypersphere," *IEEE Signal Processing Letters*, vol. 25, no. 11, pp. 1660–1664, 2018.

[8] X. Zhang, Z. Zhu, Y. Zhao, D. Chang, and J. Liu, "Seeing all from a few: 1-norm-induced discriminative prototype selection," *IEEE transactions on neural networks and learning systems*, 2018.

[9] E. Elhamifar, G. Sapiro, and S. S. Sastry, "Dissimilarity-based sparse subset selection," *IEEE TPAMI*, vol. 38, no. 11, pp. 2182–2197, 2016.

[10] I. Simon, N. Snavely, and S. M. Seitz, "Scene summarization for online image collections," in *Proceedings of the IEEE ICCV*, 2007, pp. 1–8.

[11] P. Sinha, S. Mehrotra, and R. Jain, "Summarization of personal photologs using multidimensional content and context," in *Proceedings of the 1st ACM ICMR*, 2011.

[12] H. Wang, Y. Kawahara, C. Weng, and J. Yuan, "Representative selection with structured sparsity," *Pattern Recognition*, vol. 63, pp. 268–278, 2017.

[13] H. Liu, Y. Liu, and F. Sun, "Robust exemplar extraction using structured sparse coding," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 8, pp. 1816–1821, 2015.

[14] Y. Cong, J. Liu, G. Sun, Q. You, Y. Li, and J. Luo, "Adaptive greedy dictionary selection for web media summarization," *IEEE TIP*, vol. 26, no. 1, pp. 185–195, 2017.

[15] G. Kim, L. Sigal, and E. P. Xing, "Joint summarization of large-scale collections of web images and videos for storyline reconstruction," in *Proceedings of the IEEE CVPR*, 2014, pp. 4225–4232.

[16] P. Obrador, R. De Oliveira, and N. Oliver, "Supporting personal photo storytelling for social albums," in *Proceedings of the 18th ACM MM*, 2010, pp. 561–570.

[17] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE CVPR*, 2015, pp. 1–9.

[18] P. O. Hoyer, "Non-negative sparse coding," in *Proceedings of the IEEE Workshop on NIPS*, 2002, pp. 557–565.

[19] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[20] A. Kulesza, B. Taskar *et al.*, "Determinantal point processes for machine learning," *Foundations and Trends® in Machine Learning*, vol. 5, no. 2–3, pp. 123–286, 2012.

[21] K. Zhang, W.-L. Chao, F. Sha, and K. Grauman, "Video summarization with long short-term memory," in *Proceedings of the ECCV*, 2016.

[22] A. Krizhevsky, V. Nair, and G. Hinton, "Cifar-10 and cifar-100 datasets," *URl: https://www. cs. toronto. edu/kriz/cifar. html*, vol. 6, 2009.

[23] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata, "Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly," *arXiv preprint arXiv:1707.00600*, 2017.

[24] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *IJCV*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

[25] B. Ionescu, A. L. Gînscă, B. Boteanu, M. Lupu, A. Popescu, and H. Müller, "Div150multi: A social image retrieval result diversification dataset with multitopic queries," in *MMSys*, 2016, pp. 46:1–46:6.

[26] C. Gini, "Variabilità e mutabilità," *Reprinted in Memorie di metodologica statistica. Rome: Libreria Eredi Virgilio Veschi*, 1912.