Digital Object Identifier

Summary Generation for Image Corpus using Generative Adversarial Networks

ANURAG SINGH[1] , LAKSHAY VIRMANI[1] , and A. V. SUBRAMANYAM[2], (Member, IEEE)
[1]Department of Computer Science, Netaji Subhas Institute of Technology, Delhi, India
[2]Department of Computer Science and Engineering, Indraprastha Institute of Information Technology, Delhi, India (e-mail: subramanyam@iiitd.ac.in)

Corresponding author: A. V. Subramanyam (e-mail: subramanyam@iiitd.ac.in).

ABSTRACT We propose a novel approach for image corpus 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 Generative Adversarial Network, Gini Index, Image Collection Summarization, Multi-layer Perceptron

I. INTRODUCTION

Image corpus summarization is an essential requirement for efficient representation, navigation and exploration [8]. 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 [29]. 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. Summarization of dataset can help train models without trading-off much on accuracy as the diversity of data will be maintained while saving huge computational resources. Once the appropriate hyper-parameters are known, the fine-tuning can be done over the entire original dataset.

Summarization has been well explored in videos for efficient browsing and other applications [9], [31], [32], [33], [15], though image corpus summarization has not received an equivalent attention [25]. 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. Summary of image corpus can be both qualitatively and quantitatively analyzed based on factors of relevance and diversity, which we define as,

1) *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.
2) *Diversity* maintains that all the images that are distinct are included in the summary and must not contain any redundancy [19], [21].

The main contributions of this paper are:

1) We propose a novel image corpus summarization model using MLP and GAN. Our network can be trained in an end-to-end fashion. An abstract representation is given in Fig. 1.
2) 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.
3) 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 comparing the performance against usage of original dataset. In our experiments, we demonstrate that the accuracy

FIGURE 1: The scorer gives a score for each CNN feature vector, which defines its importance for the summary. Then each feature vector, weighted by its score, is forwarded to generator for reconstruction of images. The discriminator classifies between original and summary

achieved by fine-tuning a model using the summary is comparable to the accuracy achieved when original dataset is used.

II. RELATED WORKS

Image collection summarization is an important task in multimedia processing, vision and story-telling. The summarization can be categorised into three different class of works namely, summary based on Dictionary Learning, using personal albums and using Sub-modular functions.

Dictionary Learning: In [29], Yang et. al. formulate image summarization as an optimization problem. The authors apply a dictionary learning approach based on SIFT-Bag of Words (BoW) model for creating the summary. A BoW model is used to represent the salient regions of the image collection. The summary images can be represented by these patches. On similar lines, [27] 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 [14], [1].

Personal Albums: Some works generate summary from a large personal photo albums [21], [17], [26], [10]. In [21], Sinha et. al. propose to use multdimensional features such as visual, temporal, event type, location and people. The authors also discuss the terms quality, diversity and coverage in summarization context. In [17], authors extract storyline from the album photos. The features such as face clustering and a measure of images with faces, event detection and image aesthetics ranking are used to generate the stories.

Sub-modular Functions: Sebastian et. al., in [25], solve the problem of image summarization using sub modular functions. In this work, a multiview graph is constructed to encode the contextual and temporal relation between the objects present in the input images and corresponding text. A minimum-weighted connected dominating set approach is used to select the objects. In addition, the authors in [25] also propose a novel evaluation metric V-ROUGE inspired by ROUGE, an evaluation metric extensively used in document summarization. Some other works based on this approach are [20], [16].

In order to evaluate the summary, precision and recall are usually applied. However, the precision and recall metrics define quantitatively how good a summary is only when they are provided with user annotated summary. This limits the scope to datasets where annotations are available. Further, it also adds an issue of annotating the data with relevance to a particular task for which summary is desired. It is possible that with change of task, the images needed in summary may change and hence ground truth may also change. Thus, it is quite expensive to generate different annotations for same dataset for different tasks. 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 approach has not been explored for image corpus summarization. A major advantage of our network is that it is trained in an unsupervised end-to-end manner. In the following, we describe our architecture.

III. 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 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 original dataset. The discriminator then distinguishes between the original and the reconstructed dataset. A detailed diagram is given in Figure 2.

A. 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.

B. LEARNING FRAMEWORK

In our algorithm, we first extract features of the images $\mathcal{X} = \{X_1, X_2, ..., X_n\}$ using GoogLeNet [23]. Let these features be $x = \{x_t : t = 1...n\}$. These features are fed as an input to the scorer ANN (sANN) 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 (gCNN) which reconstructs the image collection denoted by $\hat{\mathcal{X}} = \{\hat{X}_1, \hat{X}_2, ..., \hat{X}_n\}$. The discriminator (cCNN) 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. GAN

FIGURE 2: Architecture of the proposed network

in the model is similar to generative adversarial networks presented in DC-GANs and face-GANs [4]. Generator and discriminator are trained adversarially until the discriminator is not able to discriminate between the summary and original dataset.

C. 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 train the network in the following manner

1) for learning w_s, minimize $\mathcal{L}_{sparsity} + \mathcal{L}_{reconstruct} + \mathcal{L}_{task-specific}$
2) for learning w_g, minimize $\mathcal{L}_{GAN} + \mathcal{L}_{reconstruct}$
3) for learning w_d, maximize $\mathcal{L}_{GAN}$

$\mathcal{L}_{task-specific}$ is added only for the case of task-specific summarization. We explain this in more detail under task-specific loss sub-section. 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 V-B.

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
    $x \leftarrow \text{CNN}(X)$
    $S \leftarrow \text{sANN}(x)$ ▷ select frames
    $E \leftarrow xS$ ▷ weigh feature vectors by scores
    $\hat{X} \leftarrow \text{gCNN}(E)$ ▷ Reconstruction
    % Update rules:
    $w_s \overset{+}{\leftarrow} -\nabla(\mathcal{L}_{reconstruct} + \mathcal{L}_{sparsity} + \mathcal{L}_{task-specific})$
    $w_g \overset{+}{\leftarrow} -\nabla(\mathcal{L}_{reconstruct} + \mathcal{L}_{GAN})$
    $w_d \overset{+}{\leftarrow} \nabla(\mathcal{L}_{GAN})$

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.

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

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.

2) Loss of GAN $\mathcal{L}_{GAN}$

Similar to [13], we train the classifier i.e. discriminator such that it is able to distinguish between the images from the

original data set $\mathcal{X}$ and images from the reconstructed dataset by generator $\hat{\mathcal{X}}$. The $\mathcal{L}_{GAN}$ is thus defined as,

$$\mathcal{L}_{GAN} = log(\text{cCNN}(X)) + log(1 - \text{cCNN}(\hat{X})) \quad (2)$$

where cCNN(.) is the soft-max output of the discriminator.

3) Regularization loss $\mathcal{L}_{sparsity}$

This loss regularizes the number of images that form the summary. Regularization is important to ensure that the summary length is minimal. We use length regularizer loss as well as DPP loss [15],

$$\mathcal{L}_{sparsity} = \mathcal{L}_{LR} + \delta\mathcal{L}_{DPP} \quad (3)$$

where,

$$\mathcal{L}_{LR} = \left\| \frac{1}{n}\sum_{t=1}^{n} s_t - \sigma \right\|_2 \quad (4)$$

$$\mathcal{L}_{DPP} = -log(P(s_S)) \quad (5)$$

Here, σ is a hyper-parameter to control the percentage of images selected for the summary, $\delta \in \{0, 1\}$ and $\mathcal{L}_{DPP}$ is the Determinantal Point Process (DPP) loss [12]. DPP loss has also been popularly used for summarizing videos [30], [15]. 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 $D_{i,j} = s_i s_j x_i^T x_j$. And then compute the probability of the scores s_S assigned by DPP, and we have,

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

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. When $\delta = 0$, $\mathcal{L}_{sparsity} = \mathcal{L}_{LR}$, and when $\delta = 1$, $\mathcal{L}_{sparsity} = \mathcal{L}_{LR} + \mathcal{L}_{DPP}$. The value of σ, a hyper-parameter, is chosen according to the required size of the summary. Thus, by varying σ we control sparsity in terms of 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.

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

Summaries can be general or 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,

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

where $\mathcal{L}_{pre-trained}(X)$ is the loss obtained from the task specific pre-trained model and β is a hyper-parameter that scales the $\mathcal{L}_{task-specific}$. 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 V-B. The task-specific loss promotes the presence of outliers, where outliers represent both relevance and diversity. We illustrate the outliers by an example. 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, 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.

IV. 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 [11], CIFAR-100 [11], Animals with Attributes 2 (AwA2) [28], VOC2012 [2] and Diversity 2016 [7]. CIFAR-10 consists of 60,000 32X32 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 similar 60,000 32x32 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 purposes. 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.

V. EXPERIMENTS

A. IMPLEMENTATION DETAILS

We implement our architecture using TensorFlow. All the experiments are performed on Nvidia GTX 1080 GPU with 4GB RAM. It takes approximately three hours for training with 25 epochs and a batch size of 32 images. We use the output of Pool5 layer of the GoogLeNet [23] of 1024-dimensions for the feature descriptor. For our GAN model, we use DCGAN [18]. The values of the hyper-parameters

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
0.05	SUM_{gen}	0.239	0.239	0.379	0.423
	SUM_{gen}^{DPP}	0.247	0.226	0.376	0.416
	SUM_{task}	0.232	0.221	0.375	0.414
0.1	SUM_{gen}	0.223	0.215	0.335	0.365
	SUM_{gen}^{DPP}	0.215	0.199	0.328	0.363
	SUM_{task}	0.214	0.204	0.334	0.363

TABLE 2: Classification accuracy for various σ values

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
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
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
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

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 [14] is fine-tuned for the images in the corpus, and the cross-entropy loss is computed.

B. EVALUATION

The different metrics that are used to evaluate summary are reconstruction error, F-score, Gini index [3], 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 gives a general summary as it 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, and with loss $\mathcal{L}_{reconstruct}+\mathcal{L}_{LR}+\mathcal{L}_{GAN}+\mathcal{L}_{task-specific}$ as SUM_{task} and generates a task-specific summary.

1) Reconstruction Error

In [29], Yang et.al. propose that summarization is best done when the summary can reconstruct the original dataset with minimum loss. This would mean that summary concisely contains all the important parts of the data set. 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 define reconstruction error as,

$$\min \sum_{t=1}^{n} \left\| x_t - \sum_{j=1}^{\sigma n} d_j w_{jt} \right\| \tag{8}$$

where d is the feature vector which corresponds to images that are part of the summary, w_{jt} is the non-negative weight of vector d_j. We compute the weights using multiplicative algorithm in [6].

In Tables 1 and 2, we observe that the reconstruction error decreases and classification accuracy increases with increase in σ. This is expected as with increase in σ, the summary size

FIGURE 3: Figure highlighting variation of outliers and reconstruction error vs. β. The thresholds used here are 0.1 and 0.5 i.e. all images with cross-entropy loss ($\mathcal{L}_{pre-trained}$) greater than the threshold are considered to be outliers

increases and it becomes easier to reconstruct the original dataset. It is evident from the Table 1 that reconstruction error is lower, and the classification accuracy higher, when task-specific loss is incorporated. This is because minimizing the task-specific loss ensures diversity by prioritizing the inclusion of the outliers. 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 shown empirically by evaluating the number of outliers and the reconstruction error for different values of β, for the VOC2012 dataset with σ = 0.03. It can be seen in Fig. 3 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. Eventually, the error saturates as the β becomes large. We empirically find that for every dataset, there exists a value of β for which the reconstruction error is minimized as the number of outliers and inliers are balanced.

2) F-Score

Like previous works in video summarization [22], [15] where key frame annotations are given, precision and recall can be used for evaluation of summaries. Precision is the ratio of the number of overlapping images between summary and

TABLE 3: 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

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. In Table 3, precision, recall and F-score vs σ is given. 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 $\sigma = 0.5$.

3) Gini Index

In order to measure diversity, we compute Gini index [3]. For a piecewise differentiable cumulative distribution function $F(c)$ which has mean μ, and is zero for all negative values of c, Gini Index is defined as follows,

$$G = 1 - \frac{1}{\mu}\int_0^{\infty}(1 - F(c))^2 dc \tag{9}$$

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.4. 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 cannot be achieved. We observe that for summary of datasets CIFAR10, CIFAR100 and AWA2 at $\sigma = 0.01,\ 0.03$, the Gini index is best (lowest) for SUM_{task} which essentially means that the probability of choosing images from each class towards summary generation is uniform, thus maintaining the diversity. In case of VOC, we observe that SUM_{gen} gives the best Gini index.

FIGURE 4: 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 4: Classification accuracies 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%

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 4. 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

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

FIGURE 6: Ground truth summary for Diversity 2016, 1400 images (Top-20 images selected from each of 70 classes). Please zoom in for better visualization

FIGURE 7: 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

the summary captures most of the aspects of the dataset.

We also compute the accuracy when the summary is randomly sampled from original dataset. In this case, for $\sigma = 0.03$, the accuracy is 26.05%, 75.35% and 60.74% for CIFAR-100, AWA2 and VOC2012. In comparison, our SUM_{task} variant achieves 7.46%, 2.49% and 3.34% more accuracy on respective datasets. The increase in accuracy is more when the summary become sparser (smaller σ). This is expected as with more number of samples there is sufficient variance in the data to train the algorithm. Thus, the margin between the accuracies will decrease. However, the real challenge is to effectively train the network with less number of samples and in this case our algorithm samples the data while maintaining various aspects of the dataset which cannot be achieved using a random sampling.

In Tables 2 and 4, it can be observed that for the same values of σ and model variant, the accuracy is different. This is because of the fact that when the experiments are carried out for Table 2, outliers are specifically injected into the test sets. This is done in order to determine the robustness of our model against outliers. The outlier injection is also carried out in order to observe the effectiveness of the task-specific loss. We experimentally define an outlier image which has $\mathcal{L}_{pre-trained} > 0.8$ for each dataset. As no such outlier injection is carried out in the experiments for Table 4, the reported accuracies are higher in its case.

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 5. 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.

TABLE 5: 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

C. 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 dataset as well as for summaries generated by different variants. In Fig. 5c and 5d, 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. 5c. We see that the cluster is sparser compared to the same cluster highlighted in Fig. 5d. 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. 5d. In case of outliers which are a part of clusters of other classes, the task-specific loss can capture them well. In Fig. 6 and 7 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.

FIGURE 8: Generated images for CIFAR-10. First column - original images; subsequent columns show images generated from epochs 5, 10, 15, 20 and 25 respectively

In order to evaluate the image reconstruction capability of our model, we provide the following images in Fig. 8 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.

VI. DISCUSSION

In this section, we briefly explore multiple variations of the current architecture. The current study mainly focuses on task specific summary generation with respect to classification. However, summaries can be created for other tasks such as object detection, image segmentation and image captioning. For tasks like classification and object detection, SPP-net [5] can be introduced before the scorer to improve the quality of summary and also remove the necessity to either crop or re-size the images. In addition, the formulation of how outputs from pre-trained classifier (inception-v3 in our case) are added to scorer can also be investigated. For example, one of the possible variations for the task-specific loss can be,

$$\mathcal{L}_{task-specific} = \frac{(\mathcal{L}_{pre-trained}(X) - \mathcal{L}_{pre-trained}(sX))^2}{\beta} \tag{10}$$

This loss will be more sensitive to outliers. In cases where the images may be sufficiently diverse but may not be captured by the model, the loss will be very high as s will be close to zero. Thus minimizing this loss will ensure that s be high and as a consequence, the corresponding image will be present in the summary.

VII. CONCLUSION

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

REFERENCES

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

[2] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," International Journal of Computer Vision, vol. 88, no. 2, 2010, pp. 303–338.

[3] C. Gini. Concentration and dependency ratios, "Rivista di politica economica," vol. 87, 1997, pp. 769–792.

[4] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, D. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in Advances in neural information processing systems, 2014, pp. 2672–2680.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," IEEE Transactions on pattern analysis and machine intelligence, vol. 37, no. 9, 2015, pp. 1904–1916.

[6] P. O Hoyer, "Non-negative sparse coding," in Proceedings of the IEEE Workshop on Neural Networks for Signal Processing, 2002, pp. 557–565.

[7] B. Ionescu, A. Lucian GÃőnscĂČ, B. Boteanu, M. Lupu, A. Popescu, and H. MÃijller, "Div150Multi: A Social Image Retrieval Result Diversification Dataset with Multi-topic Queries," in Proceedings of the 7th International Conference on Multimedia Systems. Article 46, https://doi.org/10.1145/2910017.2910620

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

[9] A. Khosla, R. Hamid, C. Lin, and N. Sundaresan, "Largescale video summarization using web-image priors," in CVPR, 2013, pp. 2698–2705.

[13] A. Larsen, S. Sønderby, H. Larochelle, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," arXiv preprint arXiv:1512.09300, 2015.

[10] G. Kim, L. Sigal, and E. P Xing, "Joint summarization of largescale collections of web images and videos for storyline reconstruction," in CVPR, 2014, pp. 4225– 4232.

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

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

[14] 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, 2015, pp. 1816–1821.

[15] B. Mahasseni, M. Lam, and S. Todorovic, "Unsupervised video summarization with adversarial lstm networks," in CVPR, 2017, pp. 2982–2991.

[16] B. Mirzasoleiman, M. Zadimoghaddam, and A. Karbasi, "Fast distributed submodular cover: Public-private data summarization," in NIPS, 2016, pp. 3594–3602.

[17] P. Obrador, R. De Oliveira, and N. Oliver, "Supporting personal photo storytelling for social albums," in ACM MM, 2010, pp. 561–570.

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

[19] I. Simon, N. Snavely, and S. M. Seitz, "Scene summarization for online image collections," in ICCV, 2007, pp. 1–8.

[20] A. Singla, S. Tschiatschek, and A. Krause, "Noisy Submodular Maximization via Adaptive Sampling with Applications to Crowdsourced Image Collection Summarization," in AAAI, 2016, pp. 2037–2043.

[21] P. Sinha, S. Mehrotra, and R. Jain, "Summarization of personal photologs using multidimensional content and context," in ICMR, 2011.

[22] Y. Song, J. Vallmitjana, A. Stent, and A. Jaimes, "Tvsum: Summarizing web videos using titles," in CVPR, 2015, pp. 5179–5187.

[23] C. Szegedy, W> Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, et al., "Going deeper with convolutions," in CVPR, 2015, pp. 1–9.

[24] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in CVPR, 2016, 2818–2826.

[25] 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.

[26] D. Wang, T. Li, and M. Ogihara, "Generating Pictorial Storylines Via Minimum-Weight Connected Dominating Set Approximation in Multi-View Graphs," In AAAI, 2012, pp. 683–689.

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

[28] 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.

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

[30] K. Zhang, W. Chao, F. Sha, and K. Grauman, "Video summarization with long short-term memory," in Proceedings of the European conference on computer vision," 2016, pp. 766–782.

[31] B. Zhao, X. Li, and X. Lu, "HSA-RNN: Hierarchical structure-adaptive RNN for video summarization," in CVPR, 2018, 7405–7414.

[32] K. Atsushi, L. Van Gool, Y. Ushiku, and T. Harada, "Viewpoint-aware video summarization," in CVPR, 2018, 7435–7444.

[33] R. Mrigank, Y. Linwei, and SW. Yang, "Video Summarization Using fully convolutional sequence networks," arXiv preprint arXiv:1805.10538, 2018.