This CVPR Workshop paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

DEEPDISTAL : Deepfake Dataset Distillation using Active Learning

Md Shohel Rana*
Florida Gulf Coast University
Fort Myers, FL 33967, USA
mrana@fgcu.edu

Mohammad Nur Nobi
The University of Texas at San Antonio
San Antonio, TX 78249, USA
mohammadnur.nobi@utsa.edu

Andrew Sung
The University of Southern Mississippi
Hattiesburg, MS 39406, USA
andrew.sung@usm.edu

Abstract

In the rapidly evolving landscape of artificial intelligence (AI), particularly in the Deepfake domain, large-scale datasets play a pivotal role in ensuring performance, including the model's accuracy, robustness, trustworthiness, etc. However, the increasing size and intricacy of the datasets impose a growing demand for computational resources and amplify the cost and duration of model building. To mitigate the challenge, dataset distillation provides a solution. For the Deepfake detection problem, noteworthy datasets such as VDFD, FaceForensics++, DFDC, and Celeb-DF underscore the indispensability of extensive data for ensuring model robustness. Nevertheless, the computational requirement associated with these datasets presents significant obstacles. This paper describes a data distillation method utilizing Active Learning to reduce dataset size while retaining essential data qualities. The proposed method facilitates efficient model training selecting representative samples by capturing the most salient features, thereby enabling effective performance in resource-constrained environments. The study encompasses developing a data distillation algorithm tailored for Deepfake detection, rigorous experimentation with a major Deepfake dataset to validate its efficacy, and a comprehensive comparison of the model performance trained on distilled versus original datasets. Through thorough analysis, we demonstrate the practicality and effectiveness of our proposed method in alleviating computational demands without compromising detection accuracy.

1. Introduction

In artificial intelligence (AI), large-scale datasets have become the key to training accurate, robust, and bigger models (i.e., consisting of a higher number of parameters) by using larger computational resources. However, with increased samples and complexity in a dataset, training on the entire dataset can be computationally expensive and time-consuming. As a remedy, dataset distillation helps reduce the training process's computational requirements while preserving crucial information. That is— data distillation is a process of choosing a terse, high-quality data summary that enfolds the representative characteristics of the original dataset [17]. As an outcome, a distilled version of the dataset contains the target dataset's statistical properties and generalization capabilities.

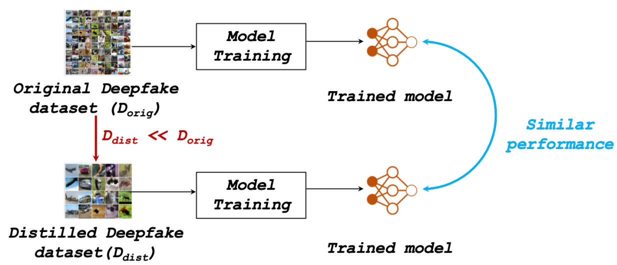

Figure 1. Overview of data distillation.

Like other computer vision domains, Deepfake tools and techniques are also data-heavy. Significant training data and computational resources are needed for Deepfake detection machine learning (ML) methods. Well-established datasets like DFDC [4], Celeb-DF [6], Faceforensics++ (FF++) [16], Versatile Deepfake Dataset (VDFD) [12], etc., are the backbone of Deepfake-related research and are very big. Even in real-world applications, each ML model must be trained on a big dataset for robustness and generalization. However, by increasing the amount of data, both upstream and downstream generalization can be possible at a faster rate [17]. In a Deepfake domain, one may have access to very limited resources for model training and inference. In other cases, computational time could be a bigger concern, even if resources are available. In both cases, having fewer training samples could potentially solve the problem. High-quality Deepfake data, yet low in size, could communicate most of the related features captured during model training. In the context of Deepfake detection [10, 11, 13, 14], the goal of data distillation includes:

- Reducing training time and resource consumption.
- Maintaining high detection accuracy with a smaller dataset.
- Ensuring the distilled dataset includes a wide variety of deepfake techniques and qualities

Designing a concise dataset summarizing samples with high-quality data is a research problem. The solution could vary depending on the domain, use case, size, and quality of the dataset, etc. Due to the nature of the domain, Deepfake detection methods are highly data-dependent. The more examples one can provide, the more features and patterns the model can capture. Therefore, selecting a summarized dataset from many samples is critical and warrants research to determine a sophisticated method. This work defines a framework to get a distilled dataset based on the dataset used for the Deepfake applications. We aim to create a smaller, representative dataset while retaining its essential characteristics. This dataset can still be used to effectively train models to distinguish between real and fake videos.

We summarize our contribution below.

- We propose a Deepfake data distillation framework to design a terse dataset summarizing samples with high-quality data to maintain high detection accuracy.
- We experiment with VDFD Deepfake datasets an emerging and widely used for Deepfake-related research.
- We compare the performance of the models trained based on the distilled and full datasets to show our proposed framework's efficacy.
- Finally, we discuss our findings, challenges, and future directions.

We organize the paper as follows. This section introduces the data distillation problem for the Deepfake framework. We summarize works related to the legacy datasets and data distillation practices in the Deepfake domain in Section 2. Section 3 proposes a Deepfake data distillation framework and defines related terminologies. Section 4 evaluates the experimented results and summarizes the findings. Finally, Section 5 concludes the paper.

2. Related Work

There has been little work in the literature on dataset distillation in the context of Deepfake. However, some works in the Deepfake area focus on knowledge distillation, which is a distinct area that tackles different issues and lies outside the scope of our current study. The center of our work remains squarely on *dataset distillation*. In this section, we aim to examine various strategies, underscoring their relevance to the broader machine learning field. We focus on the implementation and significance of dataset distillation techniques, demonstrating their essential role in driving progress within this domain.

Tongzhou et al. [20] introduce a novel approach termed "Dataset Distillation", presenting an alternative model training formulation. Instead of adjusting the model architecture, the authors propose keeping the model fixed and focusing on distilling knowledge from a large training dataset into a smaller one. This is achieved by synthesizing a small number of data points that don't necessarily adhere to the correct data distribution but approximate the performance of the model trained on the original dataset when used as training data. For instance, the authors demonstrate the capability to compress 60,000 MNIST training images into just ten synthetic distilled images (one per class), achieving close-to-original performance with minimal gradient descent steps and a fixed network initialization. The effectiveness of the proposed method is evaluated across various initialization settings and learning objectives, showing superior performance compared to alternative methods in experiments conducted on multiple datasets.

Bo et al. [22] introduced a novel approach called Dataset Condensation, designed to facilitate data-efficient learning by synthesizing a compact yet informative training set from a larger dataset. This technique condenses the original dataset into a smaller set of synthetic samples, which is suitable for training deep neural networks from scratch. The authors frame this objective as a gradient matching problem, wherein the gradients of the weights from deep neural networks trained on both the original and synthetic data are aligned and highlight its effectiveness in improving model performance while significantly reducing the size of the training dataset.

Authors [9] introduced the concept of data distillation as a means of training deep neural networks using large-scale unlabeled data. Unlike traditional supervised learning approaches that rely on labeled datasets, data distillation leverages the abundance of unlabeled data to enrich the learning process. The authors propose a methodology for distilling valuable information from unlabeled data to enhance the performance of neural networks, particularly in tasks such as object recognition and classification. Through extensive experiments, they demonstrate the efficacy of data distillation in achieving state-of-the-art results in various computer vision tasks.

Guang Li et al. [5] proposed an innovative strategy for the distillation of gastric X-ray images through the use of soft labeling and gradient descent methods, tackling the critical issues of large dataset sizes and privacy concerns in the sharing of medical data for the development of computer-aided diagnosis (CAD) systems. Departing from the conventional model distillation approach, which typically involves transferring knowledge from a larger to a smaller model, this technique seeks to generate a smaller, distilled dataset characterized by a modified data distribution. This includes the optimization of distilled images, labels, and learning rates. By doing so, it effectively minimizes the dataset's size and anonymizes the images, thereby safeguarding patient privacy.

Justin Cui et al. [3] introduced a breakthrough in Dataset Distillation, aiming to reduce large datasets into smaller, highly efficient versions for better training performance and lower storage requirements. It tackles the high memory demands of trajectory-matching-based methods (MTT) on large datasets like ImageNet-1K by proposing a method for computing unrolled gradients with constant memory, significantly reducing memory consumption. Additionally, the paper addresses MTT's difficulties with datasets having many classes through an innovative soft label assignment, enhancing convergence. This approach enables the distillation process to scale up to 50 images per class on ImageNet-1K on a single GPU, surpassing previous capabilities and achieving remarkable accuracy with fewer data points, thereby setting new standards in the field.

In theoretical analysis of machine learning, the general problem of finding a smallest subset of samples from a given dataset for model training in order to achieve a specified level of performance is an NP-hard problem (which is the same complexity as finding the smallest subset of features for model training to guarantee a minimum level of the model's performance) [15]. Therefore, heuristic algorithms are called for in dealing with the practical problem of sampling.

3. Deepfake Dataset Distillation Framework

This work focuses on formalizing the Deepfake dataset distillation framework and proposes a solution leveraging active learning. We refer to this as **Deep**fake Dataset **Dist**illation using **A**ctive **L**earning (**DEEPDISTAL**). The goal of this framework is to construct a dataset, $\mathcal{D}_{dist}$, from a finite dataset, $\mathcal{D}$, such that $\mathcal{D}_{dist} \subseteq \mathcal{D}$. $\mathcal{D}_{dist}$ represents as many features of $\mathcal{D}$ as possible and would be utilized as a substitute of $\mathcal{D}$.

The DEEPDISTAL framework has five key components such as frame extraction, frame pre-processing, feature extraction, obtaining representative frames, and, finally, obtaining representative samples ($\mathcal{D}_{dist}$). Each of these components needs to execute sequentially, as described below.

Frame Extraction. Let V be the set of all samples (videos) in a Deepfake dataset, and $\mathcal{F}$ is the set of frames extracted from V. Let $\mathcal{F}_i$ be the frames extracted from a sample (video) i. The frame extraction process can be represented as follows:

$$\mathcal{F}_i = \{f_{i1}, f_{i2}, \ldots, f_{in}\}$$

n is the total number of frames extracted from video i.

Frame Preprocessing. After frame extraction, each frame is resized to a user-defined width, W, height, and H. Let $resize(f, W, H)$ represent the resizing operation applied to a frame f. The resized frame is normalized to ensure consistent pixel values across all frames. This normalization process can be represented as:

$$f'_{ij} = \text{normalize}(\text{resize}(f_{ij}, W, H))$$

Where f_{ij} is the j-th frame from video i. Each frame is then labeled based on the video it belongs to. Let L_{ij} represent the label assigned to frame f_{ij}, where $L_{ij} = 0$ for real frames and $L_{ij} = 1$ for fake frames.

Feature Extraction. Once the frames are available, the features from each frame must be extracted to feed them into an ML model. There are many methods to obtain features from the frames [1, 7, 8, 18, 19, 21]. We propose leveraging a pre-trained deep learning (DL) model, such as VGG16, ResNet, etc., to extract features from each frame. Let X_{ij} represent the feature vector extracted from frame f_{ij}. Let ϕ denote the feature extraction function. For a prepossessed frame f_{ij}, this process can be represented as

$$X_{ij} = \phi(f_{ij})$$

Representative Frames Selection. Apparently, all the frames of set $\mathcal{F}$ do not contain significant features. We will remove non-significant and irrelevant frames from F and construct $\mathcal{F}_{rep}$ such that $\mathcal{F}_{rep} \subseteq \mathcal{F}$.

We exploit *active learning* to obtain $\mathcal{F}_{rep}$, as demonstrate in Algorithm 1. To obtain this, a function U computes the uncertainty of feature vectors. For a set of feature vectors $V = \{v_1, v_2, \ldots, v_m\}$, the uncertainty of each vector is:

$$u_i = U(v_i)$$

In active learning, each sample in the dataset is associated with uncertainty, reflecting its potential informativeness for model training. Samples with high uncertainty indicate areas where the model lacks confidence, suggesting that learning from these samples could enhance model performance significantly. This uncertainty metric guides the selection of informative data points for labeling, allowing active learning to improve model performance while reducing the need for labeled data. By prioritizing uncertain samples, active learning ensures an efficient and cost-effective learning process, particularly beneficial in scenarios with limited labeling resources or costly labeling procedures. In the process of computing uncertainties, each dataset sample is evaluated using a predictive model to determine its uncertainty level, often by deriving probabilities. This process may include measuring the distance from the decision boundary or calculating entropy. The algorithm then selects the sample with the highest uncertainty by identifying the maximum value among these uncertainty measures. This sample, indicating the greatest uncertainty, is included in the training set for subsequent rounds of training, a typical approach in active learning strategies aimed at iteratively enhancing the model with the most informative samples.

At this point, we must choose frames to ensure diversity and capture deepfake features and variations in the original dataset. We leverage the computed uncertainty score to obtain the selected frames. We choose vectors with the highest uncertainty for labeling. We aim to balance the selection across real and fake categories and combine the selected indices from both classes into a single list, S. S holds the index of selected frames across different samples that could represent the features of samples of the entire dataset. Let $\mathcal{F}_{rep}$ represent the set of representative frames mapping based on the available indices in S. Algorithm 1 discusses details on obtaining representative frames.

Representative Samples Selection. At this stage, we have a *concise* set of representative frames across *many* samples in the entire dataset as the outcome of the active learning. This representative frames set may not contain frames from all the samples. In other words, all the samples may not be equally significant and hold important features (frames). Building a model based on this representative set of frames may not provide a consistent outcome.

To overcome this problem, we rank each sample with respect to the frame frequency in the $\mathcal{F}_{rep}$ set. The higher frame count of a sample indicates the greater significance of that sample in holding Deepfake features. We rank samples according to their significance and choose samples with higher significance. As discussed in the step 6 of Algorithm 2, D_{dist} is the set of filtered K_{dist} samples obtained based on the frame frequency in $\mathcal{F}_{rep}$ and distilled dataset ratio, r. Note that finding an optimum r is a trial-and-error process. We generally start with a bigger r and gradually reduce it.

Algorithm 1 Obtain Representative Frames using Active Learning

1. Input:
   - X : Feature vectors of real and fake frames.
   - E : Total number of anticipated representative frames (E is the size of $\mathcal{F}_{rep}$).
2. Output:
   - Set of representative frames, $\mathcal{F}_{rep}$.
3. Core-set Selection:
   (a) Initializing uncertainties for each frames in X are calculated using the function U, resulting in a vector, v of uncertainties $\{v_1, v_2, \ldots, v_n\}$.
   (b) Initializing selected indices Lists using two empty lists are initialized to store the chosen indices for real and fake frames, denoted as S_{real} and S_{fake}, respectively.
   (c) Setting the number of frames to define N as the desired number of frames to be selected for each class.
   (d) Iteratively Selecting Frames:
      i. Find the frames with the highest uncertainty index: $i_{\text{max}} = \arg\max_i(u)$
      ii. Determine the label of the selected frame: $L_{i_{\text{max}}} \in \{0, 1\}$
      iii. Update the selected indices lists:
         - $S_{\text{real}} \leftarrow S_{\text{real}} \cup \{i_{\text{max}}\}$ if $L_{i_{\text{max}}} = 0$ and $|S_{\text{real}}| < N$
         - $S_{\text{fake}} \leftarrow S_{\text{fake}} \cup \{i_{\text{max}}\}$ if $L_{i_{\text{max}}} = 1$ and $|S_{\text{fake}}| < N$
      iv. Update the uncertainties: $u(i_{\text{max}}) = -\infty$
   (e) Combine the selected indices from both classes into a single list: $S = S_{\text{real}} \cup S_{\text{fake}}$
   (f) Obtain respective frames $\mathcal{F}_{rep}$ based on the combined selected indices, S:

$$\mathcal{F}_{rep} = \{X_i | i \in S\}$$

4. Evaluation

4.1. DEEPDISTAL Framework Implementation

Dataset. We deploy the DEEPDISTAL framework utilizing the (**VDFD**) [12]. The VDFD stands out as a pioneering solution that effectively overcomes significant limitations present in existing datasets such as Celeb-DF [6], DFDC [4], and FF++ [16]. By strategically addressing these shortcomings, the VDFD emerges as a refined and comprehensive resource for advancing research in Deepfake detection. The dataset sources high-quality videos from YouTube HQ, ensuring original and Deepfake content maintains exceptional visual fidelity. The dataset has thousands of samples as part of real and fake samples. Since training a model on video samples is time- and resource-consuming, we take a random set of samples and consider this to be our full dataset for this work. This experiment's full dataset holds 200 original and 200 Deepfake videos and refers to this as **Full-Set**.

DL Models Architecture and Training. To evaluate the performance of the generated distilled dataset based on the proposed DEEPDISTAL framework, we train different deep neural network models based on both full and distilled datasets. In particular, we build **Xception** and **DenseNet** DL models initialized with ImageNet pre-trained weights. We employed a transfer learning strategy, modifying the final layer to have two

Algorithm 2 Dataset Distillation

1. Input:
   - V_{real} : Path containing real samples
   - V_{fake} : Path containing fake samples
   - W, H : Target width and height for frame resizing
2. Output:
   - $\mathcal{D}_{dist}$: Distilled dataset.
3. Prepossessing:
   (a) For each sample $v_i \in V_{\text{real}}$:
      i. Extract frames F_i at regular intervals.
      ii. For each frame $f_{ij} \in F_i$:
         - Resize to $W \times H$ and normalize: $f'_{ij} = \text{normalize}(\text{resize}(f_{ij}, W, H))$
      iii. Assign label $L_{ij} = 0$ indicating real content.
   (b) For each sample $v_j \in V_{\text{fake}}$:
      i. Extract frames F_j at regular intervals.
      ii. For each frame $f_{jk} \in F_j$:
         - Resize to $W \times H$ and normalize: $f'_{jk} = \text{normalize}(\text{resize}(f_{jk}, W, H))$
      iii. Assign label $L_{jk} = 1$ indicating fake content.
4. Feature Extraction:
   (a) For each prepossessed frame f'_{ij} and f'_{jk}:
      i. Extract feature vectors using a pre-trained feature extraction function, ϕ:
         - $X_{ij} = \phi(f'_{ij})$
         - $X_{jk} = \phi(f'_{jk})$
5. Obtain representative frames, $\mathcal{F}_{rep}$:
   (a) Combine feature vectors from real and fake samples:
   $$X = \{X_{ij}\} \cup \{X_{jk}\}.$$
   (b) Apply Active Learning on X using Algorithm 1 to get $\mathcal{F}_{rep}$.
6. Obtain representative samples based on representative frames, $\mathcal{F}_{rep}$.
   (a) Get user-defined (UD) distilled dataset ratio (r) w.r.t $\mathcal{D}$. Let the size of $\mathcal{D}$ be K. The number of samples in the distilled dataset, K_{dist} is obtained subsequently, K_{dist} = r% of K.
   (b) $Hist_X$ = Get the frames frequency distribution across different samples and rank them based on their frames frequency in $\mathcal{F}_{rep}$.
   (c) Select top K_{dist} samples from $Hist_X$.
   $$D_{dist} = \bigcup_{k=1}^{K_{dist}} Hist_X$$

outputs with SoftMax activation. The training process was guided by the Greedy Layer-wise Pre-training (GLP) technique [2] Experimentation on the distilled dataset was conducted across a range of learning rates, specifically the learning rate as determined by our method, 0.002, and several epochs $3, 15$, identifying and documenting the most effective combination. If accepted, we will publish related source code in the public domain.

DeepDistAL Framework Configuration. As discussed in Section 3, there are certain user inputs and configuration of the DeepDistAL framework. First of all, for the frame preprocessing, we use 299 as both width and height. Therefore, we resize each frame to a size of 299x299. Second, as discussed, the DeepDistAL framework leverages a trained DL model for feature extraction. For this piece of work, we utilize the VGG16 trained model as the feature extraction function, ϕ. Finally, we generate two sets of distilled datasets using 50% and 20% as the ratio, r, and refer to them as **Distill−50** and **Distill−20**, respectively. Also, we choose to construct 1000 frames as the representative frames for the Distill−50 dataset and 500 frames for Distill−20.

4.2. Active Learning

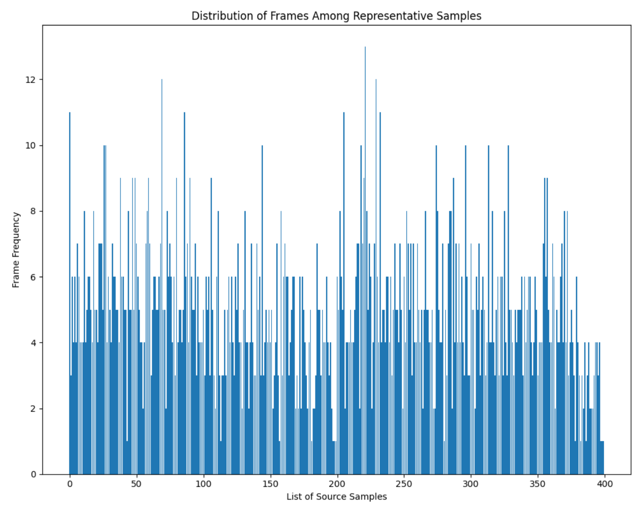

Figure 2. Representative Frames Frequency across Samples (Distill−50).

Algorithm 1 illustrates in detail how we apply active learning to a dataset. For a distilled dataset, we give a set of extracted feature vectors for the real samples, a set for fake samples, and the number of anticipated representative features, E. Choosing a good E is challenging and significantly impacts the results, as discussed in Section 4.3. As mentioned before, while generating representative frames set for Distill−50, we choose to construct 1000 frames. For the for Distill−20 dataset, we construct 500 frames.

Figure 2 and Figure 3 represent the outcome of active learning for Distill−50 and Distill−20 datasets, respectively. In each figure, the X-axis shows the list of samples, and the Y-axis represents the respective sample's frame

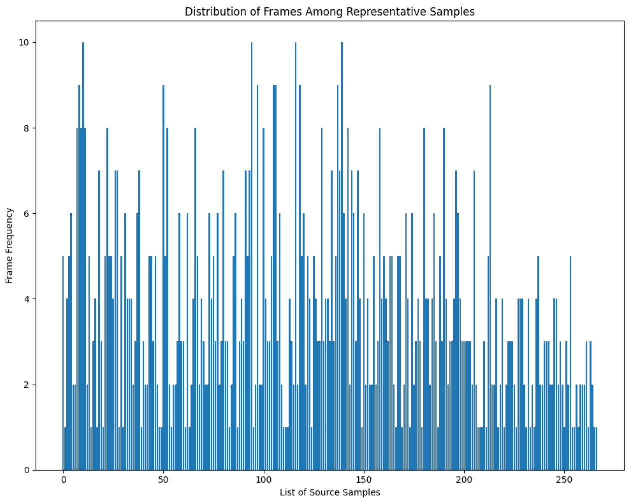

Figure 3. Representative Frames Frequency across Samples (Distill−20).

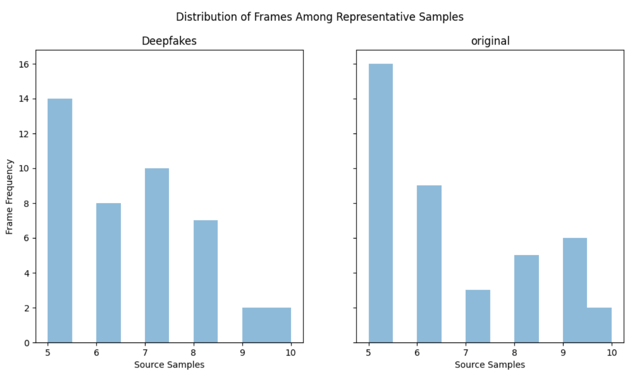

Figure 4. Frame Frequency Distribution across Deepfake and Original Samples ((Distill−20)).

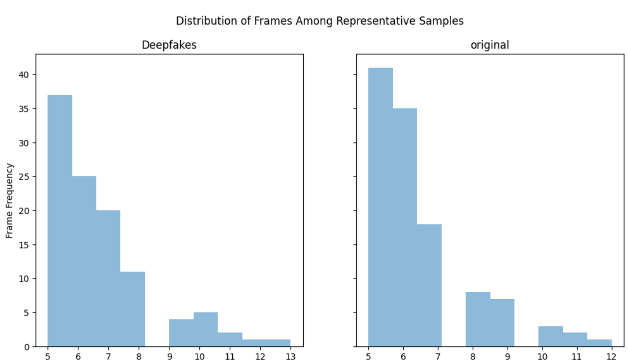

Figure 5. Frame Frequency Distribution across Deepfake and Original Samples ((Distill−50)).

count in the representative frames set. As shown in the figures, the number of frames in the representative set varies extensively from sample to sample, indicating unequal significance of samples for the Deepfake detection. We notice similar distributions once plotted in separate graphs for real and fake samples. Figure 4 and Figure 5 illustrate the real and fake samples-wise frame distribution for Distill−20 and Distill−50 datasets, respectively.

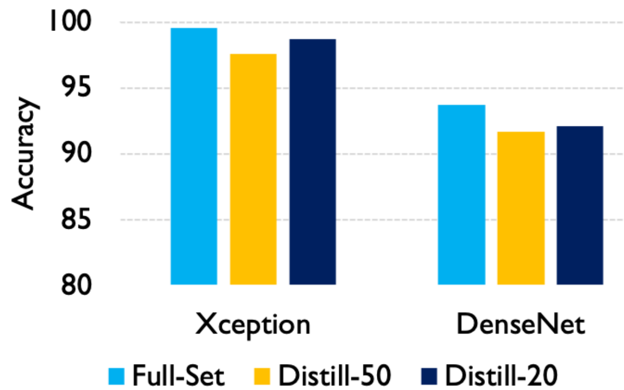

Figure 6. Performance of DL models on Distilled datasets.

4.3. Results

To the best of our knowledge, our research is the pioneering effort to concentrate on data distillation within the domain of Deepfake. As such, we were unable to draw a direct comparison with any prior work. Nevertheless, we evaluated the efficacy of models trained on distilled datasets against those trained on complete datasets.

Baselines. To compare the performance of the models trained on distilled datasets (Distill−50, and Distill−20), we train Xception and DenseNet models using the VDFD Full-Set dataset (we define the Full-Set in Section 4.1. We consider the performance of the models trained on Full-Set to be our baseline performance.

Dataset and Model Specific Insights. Figure 6 illustrates the accuracy of Xception and DenseNet models across Full-Set, Distill−50, and Distill−20 datasets. As demonstrated, the Xception model performs better overall than the DenseNet. In both cases, models trained with a Full-Set dataset have better performance. Interestingly, we observed, the models trained with Distill−50 dataset performed lower than the models trained with Distill−20 dataset. Our findings suggest that the size of representative frames in active learning played a vital role here. The dataset distilled with a lower representative frame size performed better than the distilled with a higher one.

We also measure precision, recall, and F1-score to get deeper insights across all the datasets and models. We illustrate these results in Table 1. This comparison sheds light on how dataset distillation, designed to decrease dataset size while preserving critical information, affects these models

Table 1. Performance DL models

Dataset	DL Model	Prec.	Recall	F1-Sc.	Acc.
Full-Set	Xception	99.30	99.80	99.55	99.55
	DenseNet	89.17	99.59	93.10	93.75
Distill−50	Xception	99.05	96.06	97.53	97.57
	DenseNet	86.49	99.86	92.91	91.65
Distill−20	Xception	97.60	99.94	98.75	98.74
	DenseNet	86.93	99.26	92.08	92.10

and provides deep insights into their ability to adapt and remain consistent under constrained data conditions. Xception, in particular, demonstrates exceptional resilience, with only slight variations in its performance across the varying degrees of dataset reduction. Notably, its precision and recall remain high, with recall even seeing an enhancement in the most condensed dataset scenario (Distill−20), showcasing its superior adaptability.

On the other hand, DenseNet demonstrates particular strength in maintaining high recall across all datasets, indicating its proficiency in identifying relevant instances despite the dataset size reductions. However, its precision and, consequently, F1-Score and Accuracy see more variation, particularly as the dataset undergoes distillation. This pattern suggests that while DenseNet is adept at capturing relevant instances, its precision is more susceptible to the changes in dataset composition, which might result in a higher rate of false positives as the dataset size decreases.

5. Conclusion and Future Work

This paper examines the importance and challenges associated with large-scale datasets, specifically focusing on their role in Deepfake detection. Distilling datasets into concise yet informative representations serves to reduce the computational cost while retaining essential information crucial for effective model training.

Our proposed Deepfake data distillation framework (DEEPDISTAL) provides a structured approach to designing distilled datasets tailored for high-quality model training for Deepfake detection. Through experimentation with a Deepfake dataset, we have demonstrated the efficacy of our framework in reducing computational burdens while preserving detection accuracy. Furthermore, our comparison of model performances trained on distilled versus full datasets demonstrates the practical benefits of our approach.

There are some areas for improvement in our existing work that we plan to emphasize in the future. The size of our distilled datasets is still bigger than what we aimed to achieve. In a more resource-constrained environment, one might consider tiny datasets possibly comprising 2-5% of the original dataset as most desirable. Also, we experimented with only one dataset for our current work. To have a more comprehensive analysis of the framework, it is important to experiment with multiple datasets, especially larger datasets. Finally, there are avenues to do more experiments with the parameters of the proposed framework to optimize the algorithms.

References

[1] Octavio Arriaga, Matias Valdenegro-Toro, and Paul-Gerhard Ploger. Real-time convolutional neural networks for emotion and gender classification. *ArXiv*, abs/1710.07557, 2017. 3

[2] Yoshua Bengio, Pascal Lamblin, Popovici Dan, and Hugo Larochelle. Greedy layer-wise training of deep networks. In *Advances in Neural Information Processing Systems*. MIT Press, 2006. 5

[3] Justin Cui, Ruochen Wang, Si Si, and Cho-Jui Hsieh. Scaling up dataset distillation to imagenet-1k with constant memory. In *Proceedings of the 40th International Conference on Machine Learning*. JMLR.org, 2023. 3

[4] Brian Dolhansky, Joanna Bitton, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton Ferrer. The deepfake detection challenge (dfdc) dataset. *arXiv preprint arXiv:2006.07397*, 2020. 1, 4

[5] Guang Li, Ren Togo, Takahiro Ogawa, and Miki Haseyama. Soft-label anonymous gastric x-ray image distillation. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 305–309, 2020. 2

[6] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-df: A large-scale challenging dataset for deepfake forensics. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3207–3216, 2020. 1, 4

[7] T. Prabakaran, Loveleen Kumar, S Ashabharathi, S Prabhavathi, Maneesh Vilas Deshpande, and Mochammad Fahlevi. Key frame extraction analysis based on optimized convolution neural network (ocnn) using intensity feature selection (ifs). In *2022 2nd International Conference on Technological Advancements in Computational Sciences (ICTACS)*, pages 842–847, 2022. 3

[8] Arief Bramanto Wicaksono Putra, Achmad Fanany Onnilita Gaffar, Agusma Wajiansyah, and Ismail Hasan Qasim. Feature-based video frame compression using adaptive fuzzy inference system. In *2018 International Symposium on Advanced Intelligent Informatics (SAIN)*, pages 49–55, 2018. 3

[9] Ilija Radosavovic, Piotr Dollár, Ross Girshick, Georgia Gkioxari, and Kaiming He. Data distillation: Towards omni-supervised learning. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4119–4128, 2018. 2

[10] Md Shohel Rana and Andrew H. Sung. Deepfakestack: A deep ensemble-based learning technique for deepfake detection. In *2020 7th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2020 6th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, pages 70–75, 2020. 2

[11] Md Shohel Rana and Andrew H. Sung. Deepfake detection: A tutorial. In *Proceedings of the 9th ACM International Workshop on Security and Privacy Analytics*, page 55–56, New York, NY, USA, 2023. Association for Computing Machinery. 2

[12] Md. Shohel Rana, Beddhu Murali, and Andrew H. Sung. Deepfake detection using machine learning algorithms. In *2021 10th International Congress on Advanced Applied Informatics (IIAI-AAI)*, pages 458–463, 2021. 1, 4

[13] Md Shohel Rana, Mohammad Nur Nobi, Beddhu Murali, and Andrew H. Sung. Deepfake detection: A systematic literature review. *IEEE Access*, 10:25494–25513, 2022. 2

[14] Md Shohel Rana, Md Solaiman, Charan Gudla, and Md Fahimuzzman Sohan. Deepfakes – reality under threat? In *2024 IEEE 14th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 0721–0727, 2024. 2

[15] Bernardete Ribeiro, Jose Silva, Andrew H. Sung, and Divya Suryakumar. Critical feature selection and critical sampling for data mining. In *Computational Intelligence, Cyber Security and Computational Models. Models and Techniques for Intelligent Systems and Automation*, pages 13–24, Singapore, 2018. Springer Singapore. 3

[16] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Niessner. Faceforensics++: Learning to detect manipulated facial images. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1–11, 2019. 1, 4

[17] Noveen Sachdeva and Julian McAuley. Data distillation: A survey. *arXiv preprint arXiv:2301.04272*, 2023. 1, 2

[18] K. Sugandhi, Farha Fatina Wahid, and G. Raju. Inter frame statistical feature fusion for human gait recognition. In *2019 International Conference on Data Science and Communication (IconDSC)*, pages 1–5, 2019. 3

[19] Milind Talele, Rajashree Jain, and Priti Kulkarni. Review of face emotion recognition using feature extraction techniques. In *2023 Intelligent Computing and Control for Engineering and Business Systems (ICCEBS)*, pages 1–6, 2023. 3

[20] Tongzhou Wang, Jun-Yan Zhu, Antonio Torralba, and Alexei A. Efros. Dataset distillation. *arXiv preprint arXiv:1811.10959v3*, 2020. 2

[21] Guozhen Zhang, Yuhan Zhu, Haonan Wang, Youxin Chen, Gangshan Wu, and Limin Wang. Extracting motion and appearance via inter-frame attention for efficient video frame interpolation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5682–5692, 2023. 3

[22] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. Dataset condensation with gradient matching. *arXiv preprint arXiv:2006.05929v3*, 2020. 2