Comparative Analysis of Supervised, Self-supervised, and Mixed Learning Methods for Image Content Understanding using the BAREL Approach

Aldric Pierrain[a*]

[a] Security Department, Computer Science and Information Technology, Cydral Technology, Paris, France.

KEYWORDS

Self-supervised Training
Reinforcement Learning
Knowledge Transfer
Mimetic AI
Deep Learning Network
CNN
Facial Recognition
Security

ABSTRACT

Designing large and accurate deep neural networks for visual content comprehension tasks is challenging and requires a lot of labeled training data. In this paper, we compare different supervised, self-supervised, and mixed learning methods for visual content interpretation using the same convolutional neural backbone network. We propose a new hybrid learning method, called BAREL, that combines a self-supervised learning phase with a supervised reinforcement training to improve performance while reducing data labeling effort. Our experiments show that BAREL outperforms both supervised and self-supervised learning methods alone and suggest that BAREL has the potential to provide a more realistic and human-like learning approach for visual content understanding.

1 INTRODUCTION

Machine learning is a vibrant and active research area that aims at developing algorithms that can learn from data and perform various tasks such as a classification. However, most machine learning methods can be divided into two broad categories: supervised and unsupervised. Supervised methods require labeled data to train a model that can predict classes for new data. In contrast, unsupervised methods do not rely on labels but try to discover patterns or structures in data. Both approaches have their advantages and limitations[1], supervised methods often suffer from high annotation costs and lack of generalization[2]. Unsupervised methods struggle with defining meaningful objectives and evaluating their performance[3]. Therefore, there is a need for novel machine learning paradigms that can overcome these challenges and leverage both labeled and unlabeled data in an effective way.

An important contribution to this paradigm shift is the so-called generative adversarial networks (GANs)[4], which consist of two competing models: a generator and a discriminator. The generator tries to produce realistic samples from a latent space, while the discriminator tries to distinguish between real samples from data and fake samples from generator. The two models are trained in an adversarial manner until they reach an equilibrium where they cannot fool each other anymore. GANs have shown remarkable results in various domains such as image synthesis[5], image restoration[6] and text generation[7-8], to mention only these concrete applications. However, GANs are not easily tailored to a classification task or, more specifically, to the modeling of acquired knowledge and the generation of a "digital signature" at the output of its network, reflecting the stimulus response of the network in the presence of external inputs. This limits their applicability for tasks that require extracting meaningful information or features from data.

However, this dual operation mode is an insightful approach to develop a learning model based on the participation of two entities, one like the GAN discriminator that can produce a neural prewiring from the learning, resulting from a generalization focused to the long-term embedding of the knowledge, the other by reweighting the neural structure, through a phase of acquiring ground truth knowledge. Based on recent neuroscience results[9-10-11-12], we have thus tried to introduce a new learning process, unfolding two learning phases that mix self-supervised acquisition and supervised reinforcement: a) a first neural activity for an upstream learning phase aiming to establish a primary knowledge and to pre-structure the weight distribution of a specialized network, and b) a reinforcement for a knowledge consolidation phase by strictly reusing the same backbone but letting the network consolidate its inner parameters based on formal assumptions. This method attempts to mimic the behavior of the human brain which, although it apparently has some innate brain circuits[13], uses acquired knowledge to strengthen its synaptic connections issuing validated assumptions and taking advantage of "complementary facts" to accelerate the process of adjusting synaptic weights[14-15]. The method we propose aims at training a neural network with a large set of unlabeled input data in order to preset a data model, and to use in a subsequent phase a smaller set of labeled data in order to optimize the neural networks and to improve the generalization (or specialization) of the acquired knowledge. This approach is likely to outperform current learning methods while minimizing the need to create a well-designed knowledge corpus. The method we propose intends to improve memory traces with only a few predefined markers, leveraging general concepts and specific samples for specialization.

In this paper, we thus focused on comparing different learning methods for visual content interpretation, with a particular emphasis on a new hybrid learning method called BAREL for "Backpropagation And Reinforcement-based Environment for Learning". We start by reviewing results using state-of-the-art supervised and self-supervised learning methods for visual content comprehension. We then describe the BAREL method, which combines self-supervised learning with a supervised reinforcement training stage, using a comprehensive knowledge to improve performance for a specific operation.

Our experiments show that BAREL outperforms both supervised and self-supervised learning methods alone. Our

Internal Reference: PUB-20230403-CTY-I2A – April 3rd, 2023
(*) Email Address: cydraltechnology@{gmail.com}

results suggest that BAREL has the potential to provide a more realistic and human-like learning approach for visual content understanding, while requiring less data labeling effort. We believe that our study can contribute to advancing the area of computer vision and artificial intelligence and opening up new opportunities for research and applications, especially for the domains where acquiring labeled data is challenging (e.g., health). Our main activity being IT Security, the practical use case considered hereafter is facial recognition.

2 RELATED WORK

For the backbone modeling the neural network specialized in the image interpretation, we have chosen a convolution-based network and more specifically, two convolutional network types: ResNet[16] and DenseNet[17].

A convolutional network is composed of several layers that apply linear and nonlinear operations on the input data; a layer can be defined by the following formula:

$$y_{i,j,k} = \sum_m \sum_n \sum_l x_{i+m-1,j+n-1,l}\, w_{m,n,l,k} + b_k \quad (1)$$

where x is the input tensor, w is the kernel or filter weights, b is the bias term and y is the output of the layer. The indices (i, j, k) represent the spatial location and depth of the output feature map. The indices (m, n, l) represent the spatial location and depth of the kernel.

2.1 Backbones

ResNet (He et al., 2016) introduced the concept of residual learning, which enables the training of very deep networks by adding shortcut connections that skip one or more layers. This approach mitigates the vanishing gradient problem and improves the gradient flow throughout the network, leading to better convergence and accuracy. ResNet has achieved state-of-the-art results on several image classification benchmarks, including the ImageNet dataset (Russakovsky et al., 2015).

DenseNet (Huang et al., 2017) is another convolutional backbone network that adopts a different approach to deep learning. DenseNet connects each layer to every subsequent layer in a feed-forward manner, forming dense connections that allow the network to access features from all previous layers. This architecture encourages feature reuse and information flow, leading to more compact and accurate networks. DenseNet has also shown competitive results on various image classification benchmarks[18-19]. Despite its higher complexity and resource requirements, we have decided to choose DenseNet as the reference backbone for our final assessment, as it is a good feature extractor for various computer vision tasks that rely on convolutional features, including facial recognition. We indeed believe that DenseNet is closer to the human neural network structure, particularly due to its ability to form dense connections between different layers.

2.2 Metric learning techniques

For facial recognition tasks, one common approach is to use a metric learning technique called Triplet loss[20]. Triplet loss aims to learn a mapping from the input space to an embedding space, where similar faces are clustered together, and dissimilar faces are pushed apart. To achieve this, Triplet loss uses triplets of images: an anchor image, a positive image (with the same identity as the anchor), and a negative image (with a different identity).

The Triplet loss function can be formulated as follows:

$$L = \sum_{i=1}^{N} \max\left(0,\ d\left(x_i^a, x_i^p\right) - d\left(x_i^a, x_i^n\right) + \alpha\right) \quad (2)$$

where $\left(x_i^a, x_i^p\right)$ and x_i^n represent the embeddings of the anchor, positive, and negative images, respectively, for the i-th triplet. The distance metric $d(\cdot)$ measures the dissimilarity between two embeddings, such as the Euclidean or the cosine distances. The margin α controls the minimum difference between the distances of positive and negative pairs, and the max function ensures that the loss is non-negative.

The goal of this metric is to learn a CNN that can produce discriminative embeddings for each face image, such that the distance between embeddings of the same identity (positive pairs) is minimized, while the distance between embeddings of different identities (negative pairs) is maximized. This can be achieved by minimizing the Triplet loss function with respect to the network parameters. We used a Triplet loss head function to train both ResNet and DenseNet on the same facial recognition task.

In addition to supervised learning with Triplet loss, we also explore the use of Self-Supervised Learning (SSL) for visual content interpretation. SSL is a type of self-supervised learning that learns representations from unannotated data, without requiring any explicit labeling effort. One popular SSL method is Barlow Twins (Zbontar et al., 2021)[21], which aims to learn representations that maximize the agreement between two randomly augmented views of the same input image, while minimizing the agreement between different input images. The Barlow Twins loss function can be formulated as follows:

$$L = \frac{1}{D}\sum_{i=1}^{D}(1 - C_{ii})^2 + \lambda \sum_{i=1}^{D}\sum_{j=1, j\neq i}^{D} C_{ij}^2 \quad (3)$$

where L is the loss, C is the cross-correlation matrix between the normalized vectors of two views of each image, D is the dimensionality of these vectors and λ is a hyperparameter that controls the trade-off between invariance and redundancy reduction. The first term of this loss function encourages similar representations between distorted variations (augmented views) of a sample by minimizing the difference between diagonal elements of C and 1. The second term encourages diversity among learned representations by minimizing off-diagonal elements of C.

The BAREL approach combines the strengths of supervised Triplet loss and self-supervised Barlow Twins metrics. Triplet loss excels at learning task-specific features while Barlow Twins can refine the representation by reducing redundancy and improving generalization. Enabling the same backbone during the whole two-phase training process, we might achieve better performance with less (labeled) data: the network is first trained with Barlow Twins and then fine-tuned with Triplet loss. The general learning process based on (2) and (3) we propose is finally described by:

Let f be the Resnet backbone we selected, g be the projection head, x_i be an input image and y_i be its label. Let also $\mathcal{LBT}$ *be* the Barlow Twins loss function and $\mathcal{LML}$ be the metric loss function. Then,

Step 1: Train f and g for SSL using Barlow Twins

$$\min_{f,g} \mathcal{L}\, BT\left(g\left(f(x_i)\right), g\left(f(\tilde{x}i)\right)\right)$$

Step 2: Transfer f to a new task with labeled data. Let g' the new head projector associated to the updated model for the specialization. Freeze f and only update g'.

$$\min_{g} \mathcal{L}\, ML\left(g'\left(f(x_i)\right), y_i\right)$$

where $\mathcal{L}ML$ could be at final a contrastive loss, a triplet loss (as previously stated for the facial recognition use case), etc.

Relating to the second step, the mechanism is indeed based on "knowledge transfer" using a pre-trained model and adapting it to a new task or domain. To do this, we have to keep the backbone of the pre-trained model (which contains most of the parameters and features) and replace the last layers (which are specific to the target task or domain) with new ones. The backbone is denoted by $f(x;\theta)$, where x is the input image and θ are the fixed parameters from the pre-trained model. The last layers are denoted by ϕ, which are the new parameters that we want to learn, to specialize the model, using a loss function that measures how well our new model performs on the new task or domain. The loss function consists of two parts:

a) The first part is the metric loss function, denoted by L_{metric}, which compares how similar the features from the backbone of our new model are to the true labels of our images. For instance, if we want to classify images into different categories, we can use cross-entropy loss as the new metric loss function.

b) The second part is the weight decay loss, denoted by L_{wd}, which penalizes large values of ϕ to prevent overfitting. For example, we can use L2-norm as the new weight decay loss function. We also use a coefficient λ that controls how much we care about each part of the loss function. A larger λ means we care more about weight decay and less about metric loss, and vice versa.

The formula for our transfer learning loss function is then given by:

$$\min_{\phi} L\,(x, y; \phi, \theta) = \min_{\phi}[L_{metric}(f(x;\theta), y) + \lambda \cdot L_{wd}(\phi)]$$

We fine-tune the last layers of the model using stochastic gradient descent (SGD) with a small learning rate α as follows:

$$\phi' \leftarrow \phi - \alpha \nabla_{\phi} \mathcal{L}(x, y; \phi, \theta)$$

where ϕ is the model parameters, α is the learning rate, $\mathcal{L}$ is the loss function, x is the input data, y is the ground-truth labels, and $\nabla_{\phi}\mathcal{L}$ is the gradient of the loss function with respect to the model parameters ϕ. The notation ϕ' represents the updated parameters.

3 METHODOLOGY

The creation of a large and diverse database of facial images is usually essential for a supervised learning phase. We first used automated crawlers to gather people images from various online sources, such as social media and image sharing platforms. However, to ensure the diversity of our database, we also manually selected images from different publicly available datasets, including FaceScrub, CelebA, and others. To avoid any potential biases and ensure the fairness of the evaluation, we took particular care to avoid any overlap between our database and the Labeled Faces in the Wild (LFW) benchmark dataset. To achieve this, we thoroughly checked for any potential duplicates and excluded them from our database, as well as cross-checked the identities with the ones present in LFW. All images were automatically aligned using facial landscape location and cropped to 150x150 pixels.

3.1 Dataset Collection for Supervised Learning

To further increase the diversity and robustness of our database, we dynamically applied various augmentation techniques, such as zooming, flipping, color perturbation and grayscale conversion, to each image. This helped to ensure that the network is capable of recognizing faces from different angles and in different lighting conditions.

Table 1-Dataset Information for Supervised Learning.

Standard database (A) – Diverse faces	Value
Ethnicity balance	60+ countries
Age range	18-70 years
Number of individuals	7,452
Average images per individual	54
Number of images	398,391

After gathering the images, we applied automatic processes to review and curate the dataset to ensure a balanced representation across different genders and ethnicities. This also involved discarding images that did not meet certain quality standards, such as blurriness, poor lighting, or low resolution, as well as removing duplicates and irrelevant images. Faces are extracted from images using the Histogram of Oriented Gradients (HOG) method as described in Dalal N. & Triggs B. (2005)[22]. All the extracted faces from the images are normalized to a size of 150x150 pixels. No padding was added around the extracted picture to limit the impact of the face background.

3.2 Dataset Collection for Self-supervised Learning

For our self-supervised learning dataset, we employed StyleGAN[23], a generative adversarial network introduced by Karras T. et al. in their paper "A Style-Based Generator Architecture for Generative Adversarial Networks", to generate synthetic images of non-human subjects. Due to its ability to produce realistic and high-quality images, StyleGAN has become a popular choice for this type of task, and at the same time quickly meet the challenge of setting up a large learning base, with sufficient diversity and no usage rights constraint.

Table 2-Dataset Information for Self-supervised Learning.

Generated database (B) – Synthetic faces	Value
Age range	2-80 years
Number of individuals	192,225
Images per individual (with static augmentation)	1~20

Table 3-Generated realistic face of non-existent people.

Regarding this computer-generated face dataset, each person is represented by one unique face. We however made two groups

from the database: one has 20 different versions of each face; the other group is for the self-learning. We also plan to explore how changing the hidden input for the generator, to produce different angles of the same face, as a future update of the current evaluation.

3.3 Implementation Framework

For this study, we used the Dlib library[24-25-26], a general purpose cross-platform open source software library written in C++, to implement convolutional neural networks (CNNs) for face detection and recognition. Dlib provides a high-level interface for building and training deep neural networks, as well as low-level access to the underlying data structures and algorithms. Dlib also offers a set of tools for face alignment and extraction, which we used as a preprocessing step for our evaluation. This library is widely used in both industry and academia and is a popular and effective tool for image processing, especially for face-related tasks.

We used the results reported by Davis E. King, the creator of Dlib, as a reference to compare and validate our own evaluations for the facial recognition model that we developed using the BAREL approach. King D. E. shared the details of the training dataset, provided the CNN model as a default component for Dlib and based on this model, obtained some of the highest scores on the LFW benchmark. For instance, his source database contains 7,485 individuals and 3 million unique images.

To ensure a fair comparison with Dlib, we defined our own databases to have an equivalent number of unique individuals, and we mostly took the definition of the original Dlib model and trained it with our databases. Our datasets intentionally contain far fewer images than those used by this library:

Table 4-Face Verification on Labeled Faces in the Wild.

Rank	Model	Accuracy	Year
1	VarGFaceNet	99.85%	2019
	…		
17	**Dlib**	**99.38%**	**2017**
18	Light CNN-29	99.33%	2015

4 RESULTS

Our study was carried out using a cluster with several computing nodes integrating Nvidia RTX GPU cards. However, it still took two days of intensive computing for each of the models we tested.

For training the standard networks, we chose the most appropriate database, namely database (A), which was specifically used in the fine-tuning process for all the models. Database (B) was selected as the source during self-supervised training. It should be noted that (B) was pre-generated using a GAN, with a static image augmentation process embedded in the training.

To maintain consistency in the latent space dimension with the output of standard networks, the output of each model trained using the SSL method is substituted with a fusion projection of same dimensionality.

The table below displays the number of parameters implemented per model, providing an indication of their intrinsic complexity. The ResNet network adheres to a standard configuration with a 34-layer topology. On the other hand, the DenseNet reference model has been revised using a 101-layer convolutional network to reduce its complexity to a level compatible with training on basic graphics cards currently available on the market.

Table 5-Model Parameters.

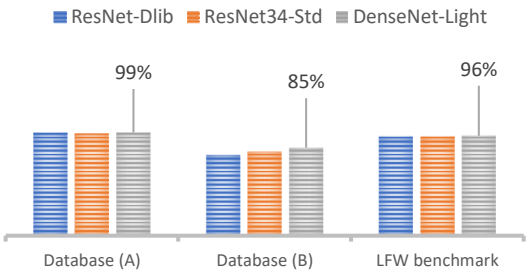
Model	Layers	Parameters
ResNet-Dlib	132	5,611,040
ResNet34-Std	152	21,354,432
DenseNet-Light	283	620,130

It can be noted that the model initially implemented by Dlib is the less complex regarding the total number of parameters; for the two new complementary models, we empirically determined a satisfactory balance for the pretext task between depth and density, and finally retained a definition with a complexity level that is certainly higher than the basic model, but nevertheless limited.

Table 6-General Evaluation Results.

Model	Inner Acc.(*2) (%)		LFW benchmark	
	DB (A)	DB (B)	Acc.	σ
Standard Dlib(*1)	99.92	92.53	98.20	6.52e-3
ResNet-Dlib ⇏ BAREL	98.85	77.40	95.17	10.45e-3
ResNet-Dlib	98.17	77.65	94.77	11.31e-3
ResNet34-Std ⇏ BAREL	98.41	80.91	95.55	12.57e-3
ResNet34-Std	97.59	74.38	94.38	13.39e-3
DenseNet-Light ⇏ BAREL	**99.29**	**84.61**	**96.10**	**10.07e-3**
DenseNet-Light	97.94	87.49	94.97	8.38e-3

*(*1) Normalized results (jitter parameter preset to 10 and similarity threshold to 0.55 for instance) for comparable assessment with the other networks.*
*(*2) Values obtained by averaging the outcomes of the "Top-1~Top-5" benchmarks applied to the datasets.*



To apply our BAREL approach, we migrated each pretrained backbone, initially trained for a pretext task, to an alternate neural network with an identical structure, excluding the projection layer tailored for task specialization. We progressively unfroze the internal layers and updated their weights using a low learning rate, specifically focusing on the optimization of the fully connected (FC) layers for the downstream task.

The Dlib library obviously has a large and probably diverse collection of images for facial recognition. The training dataset used for the Dlib model additionally introduces similar biases as the LFW tests, which is not necessarily the case with our own datasets aimed at covering a wider diversity of individuals. Our new ResNet and DenseNet models have of course more parameters than the Dlib CNN, but they suffer from insufficient training data in terms of the total number of images. Combining our two databases for a single training phase does not seem to enhance anything, instead it appears to lower the models' performance slightly. The diversity of facial captures is of course

important for learning and generalizing facial features. The best average results are seemingly achieved by the process that integrates learning and refinement in a definitive manner.

Our BAREL learning process produced some interesting effects. The first one pertains to the improvement of knowledge integration within the backbone network. The recognition scores of individuals within the training corpus, irrespective of the database used, is higher than or equivalent to those of the standard CNNs. The combination of both learning modes to initialize (pre-weighted mechanism) a neural network and enable a quick specialization exhibits its full potential here. The second remarkable effect is the performance achieved in the LFW evaluation. Even though Dlib remains the best performer, we obtained satisfactory results with BAREL without increasing either the neural complexity or the training dataset volume, which remains 13 times smaller than that used by Dlib. Our DenseNet reference model performs well, with an acceptable score on the LFW benchmark, and in fact much superior to the similarly structured but standard-trained model.

5 FURTHER DISCUSSION

Future research could investigate the use of StyleGAN or other generative models to produce more diverse and realistic facial images, with a special emphasis on varying pose and orientation. By altering the latent space and input noise of the generator, it may be indeed possible to introduce variations in facial orientation and expression that could enhance the diversity and richness of the training data, leading to better generalization performance in facial recognition tasks.

We are also investigating the potential application of the BAREL approach to another task: qualifying the visual content of an image. However, the development of a model capable of accurately determining the visual similarity between two images remains a significant challenge. The process of establishing a visual associative link between two images is complex and it is difficult to determine how to measure visual similarity. Despite this challenge, we believe that the BAREL approach, with the help of a refinement process to guide the concept of "visual similarity" based on an internal similarity feature we already internally developed, could produce promising results for this other task.

6 CONCLUSION

In this study, we focused on the positive effect of combining two learning processes to improve performance in a complex recognition task (faces). We first used an initial learning process, probably similar to our own way of acquiring prior knowledge and generalizing concepts, to provide a backbone with a pre-weighted network. The same backbone (knowledge transfer) was then exposed to a specialization phase using another learning method to further improve the performance of the neural network while drastically reducing the amount of training labeled data required.

Comparative tests with common CNNs and standard supervised or self-supervised learning phases demonstrate the benefits of the BAREL approach, which can be considered as a semi-supervised learning method. The performance evaluation of the facial recognition task was benchmarked against the LFW dataset to assess the effectiveness of this approach. Although our objective was not to achieve a higher score than our baseline mode, the evaluation results showcased that the BAREL method outperformed the traditional learning methods, achieving higher accuracy rates and reducing the required amount of training data.

Our study shows that combining different learning processes seems to be a key factor for enhancing AI understanding across different modalities. We also find that using a small unqualified knowledge base to bootstrap learning leads to better performance than relying on large but potentially noisy labeled datasets. This suggests that deep neural networks can benefit from incorporating prior knowledge into their architectures. Future work should investigate how to optimize the BAREL process as described, including how to introduce multiple successive iterations (by perhaps more closely emulating competitive GAN-type models), and how it affects generalization-specialization and robustness of AI models whatever the learning purpose.

References

[1] Murphy K. P. (2012). Machine learning: A probabilistic perspective. MIT press

[2] Zhu X. (2005). Semi-supervised learning literature survey. University of Wisconsin-Madison

[3] Bengio Y., Courville A. & Vincent P. (2013). Representation learning: A review and new perspectives. IEEE transactions on pattern analysis and machine intelligence, 35(8), 1798-1828

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

[5] Karras T., Laine S. & Aila T.A. (2019). Style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 4401-4410)

[6] Ledig C., Theis L., Huszar F., Caballero J., Aitken A., Tejani A., Totz J., Wang Z. et al. Photo-realistic single image super-resolution using a generative adversarial network. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4681-4690).

[7] Goodfellow Ian J., Pouget-Abadie J., Mirza M., Xu B., Warde-Farley D., Ozair S., Courville A. & Bengio Y. (2014). Generative adversarial networks [Preprint]. arXiv:1406.2661

[8] Donahue D. & Rumshisky A. (2019). Adversarial text generation without reinforcement learning [Preprint]. arXiv:1810.06640v2

[9] Hassabis D., Kumaran D., Summerfield C. & Botvinick M. (2017). Neuroscience-inspired artificial intelligence. Neuron, 95(2), 245-258

[10] Sussillo D. (2019). How AI and neuroscience drive each other forwards. Nature, 571(6362), 183-185

[11] Pavlick E. (2023). How does ChatGPT differ from human intelligence? Nextgov. Retrieved from

https://www.nextgov.com/ideas/2023/03/how-does-chatgpt-differ-human-intelligence/172839/

[12] Dehaene S., Lau H. & Kouider S. (2017). What is consciousness and could machines have it? Science 358(6362):486-492

[13] Serre T. (2020). The case for innate neural wiring in the human brain. Current Opinion in Neurobiology, 60, 118-125

[14] Liang J., Wang X. & Wang Y. (2021). Detecting synaptic connections in neural systems using compressive sensing and special data processing. Cognitive Neurodynamics

[15] Lewis P.A., Knoblich G. & Poe G. (2018). How memory replay in sleep boosts creative problem-solving. Trends in Cognitive Sciences 22(6):491-503

[16] He K., Zhang X., Ren S. & Sun J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778)

[17] Huang G., Liu Z., Van Der Maaten L. & Weinberger K.Q. (2017). Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4700-4708)

[18] Zhang Z., Liu Y. & Liang J. (2020). ResNet or DenseNet? Introducing dense shortcuts to ResNet [Preprint]. arXiv:2003.08941

[19] Khandelwal S. & Khandelwal A. (2021). Comparison of the performances of DenseNet-121 and ResNet-50 with different loss functions for COVID-19 detection using chest X-ray images. Informatics in Medicine Unlocked 23:100579

[20] Schroff F., Kalenichenko D. & Philbin J. (2015). FaceNet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 815-823)

[21] Zbontar J., Jing L., Misra I., LeCun Y. & Deny S. (2021). Barlow twins: Self-supervised learning via redundancy reduction. In Proceedings of Machine Learning Research 139:1502–1514

[22] Dalal N. & Triggs B. (2005). Histograms of oriented gradients for human detection. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Vol. 1 pp. 886-893)

[23] Karras T., Laine S. & Aila T.A. (2019). Style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 4401-4410)

[24] King D. E. (2009). Dlib-ml: A machine learning toolkit. Journal of Machine Learning Research 10(Jul):1755–1758

[25] Find more information about Dlib on its official website at http://dlib.net/. Also access the source code and documentation of Dlib on its GitHub repository: https://github.com/davisking/dlib

[26] Li Z., Zhang Y., & Wang J. (2021). Implementation of Dlib deep learning face recognition technology [Preprint]. arXiv:2108.13353