Intelligent Systems with Applications 25 (2025) 200488

Contents lists available at ScienceDirect

Intelligent Systems with Applications

journal homepage: www.journals.elsevier.com/intelligent-systems-with-applications

Employing the concept of stacking ensemble learning to generate deep dream images using multiple CNN variants

Lafta Alkhazraji [a], Ayad R. Abbas [b], Abeer S. Jamil [c], Zahraa Saddi Kadhim [a], Wissam Alkhazraji [a], Sabah Abdulazeez Jebur [d], Bassam Noori Shaker [e], Mohammed Abdallazez Mohammed [f], Mohanad A. Mohammed [g], Basim Mohammed Al-Araji [h], Abdulkareem Z. Mohmmed [i], Wasiq Khan [j], Bilal Khan [k,*], Abir Jaafar Hussain [l]

[a] Department of Computer Engineering Techniques, Imam Ja'afar Al-Sadiq University, Iraq
[b] Department of Computer Science, University of Technology, Baghdad, Iraq
[c] Department of Computer Technology Engineering, Al-Mansour University College, Baghdad, Iraq
[d] Department of Computer Techniques Engineering, Imam Alkadhim University College, Baghdad, Iraq
[e] Computer Science Department, College of Computer Science and Information Technology, University of Al-Qadisiyah, Al Diwaniyah, Iraq
[f] University of Karbala, College of Computer Science and Information Technology, Computer Science Department, Iraq
[g] Computer Science Department, University of Technology, Baghdad, Iraq
[h] Imam Ja'afar Al-Sadiq University, Iraq
[i] Babylon Education Directorate, Ministry of Education, Babylon, Iraq
[j] School of Computer Science and Mathematics, Liverpool John Moores University, UK
[k] School of Computer and Engineering, California State University San Bernardino, USA
[l] Department of Electrical Engineering, University of Sharjah, Sharjah, UAE

ARTICLE INFO

Keywords:
Deep dream
Stacking ensemble
Imaginary hallucinations
Inception
Xception
Deep learning

ABSTRACT

Addiction and adverse effects resulting from schizophrenia are rapidly becoming a global issue, necessitating the development of advanced approaches that can provide support to psychiatrists and psychologists to understand and replicate the hallucinations and imagery experienced by patients. Such approaches can also be useful for promoting interest in human artwork, particularly surrealist images. Accordingly, in the present, a stacking ensemble Deep Dream model was developed that aids psychiatrists and psychologists in addressing the challenge of mimicking hallucinations. The dream-like images generated in the present study possess an aesthetic quality reminiscent of surrealist art. For model development, a series of five pre-trained Convolutional Neural Network (CNN) architectures—VGG-19, Inception v3, VGG-16, Inception-ResNet-V2, and Xception were stacked in an ensemble learning approach to create Deep Dream images whereby the upper hidden layers of the architectures were activated, and the models were trained via the Adam optimizer. Performance of the proposed model was evaluated across three octaves to amplify the maximum possible patterns and features of the base image. The resulting dream-like images contain shapes that reflect elements from the ImageNet dataset on which the above pre-trained models were trained. Each of the base images was manipulated to generate various dreamed images, each one with three octaves, which were finally combined to construct the final image with its loss. Final Deep Dream image showed a loss of 47.5821, while still retaining some features from the base image.

1. Introduction

In recent years, addiction and the resulting adverse impacts of schizophrenia have increased rapidly, posing a significant challenge on multiple fronts including societal and healthcare related issues across the globe. The World Health Organization (WHO) has identified over 24

* Corresponding author.
E-mail addresses: lafta_raheem@ijsu.edu.iq (L. Alkhazraji), ayad.r.abbas@uotechnology.edu.iq (A.R. Abbas), abeer.salim@muc.edu.iq (A.S. Jamil), Zahraa_saadi@ijsu.edu.iq (Z.S. Kadhim), wissam_abbas@ijsu.edu.iq (W. Alkhazraji), sabah.abdulazeez@iku.edu.iq (S.A. Jebur), bassamsat@qu.edu.iq (B.N. Shaker), mohammed.abdallazez@uokerbala.edu.iq (M.A. Mohammed), Mohanad_ali1986@yahoo.com (M.A. Mohammed), basim.mohammed.h@ijsu.edu.iq (B.M. Al-Araji), abducsphone2@gmail.com (A.Z. Mohmmed), w.khan@ljmu.ac.uk (W. Khan), bilal.khan@csusb.edu (B. Khan), abir.hussain@sharjah.ac.ae (A.J. Hussain).

https://doi.org/10.1016/j.iswa.2025.200488
Received 15 October 2024; Received in revised form 6 January 2025; Accepted 26 January 2025
Available online 29 January 2025

million individuals suffering from schizophrenia, accounting for about 0.32 % of the global population (Schizophrenia World Human Organization (WHO) 2024). Additionally, as per the observation by the WHO indicated from 2021, psychoactive drugs have been used by approximately 296 million individuals, leading to addiction indicating 5.8 % of the global population being addicted (World Human Organization (WHO) Drugs (psychoactive) 2024). Surrealist images have recently attracted large audience with their unconventional shapes, distortions, and patterns that occasionally alter features to a degree where the original image becomes difficult to recognize (Apollinaire & Manifesto, 2023).

Deep learning in the last two decades has made significant advancements and applications in diverse sets of disciplines (Ali et al., 2023; Hazar et al., 2020). The aim of Deep Dream is to generate images that simulate the imaginations of addicted individuals and schizophrenic patients in an attempt to reflect hallucinations and illusory visions. The outcomes of the knowledge extracted from these images is typically to handle complex cases of mental salubrity and enrich practitioners who are confronted with these issues (Rastelli et al., 2022). Deep Dream images are generated via a generative CNN architecture which is fed with a source image, and features of noticeable significance are then extracted via the trained Deep Dream model (Al-Khazraji et al., 2023). In a deep CNN architecture developed for Dream images, earlier hidden layers are typically tasked with the identification of linear edges (also called low-level features) whereas the deeper layers identify complex non-linear features by utilizing low-level features from earlier layers as input. The aggregation of low and high-level features by the model are combined to create effects such as trees or an entire structure (Al-Khazraji et al., 2023). Various CNN network structures for this task have been proposed in the literature sharing several key characteristics: most have been trained on a large ImageNet dataset similar network structure (number of layers, number of units in each layer). Despite these similarities, these CNN architectures have been reported to extract distinct features, each analyzing the input image from a unique perspective (Ali et al., 2022). As a result, different CNN architectures yield distinct features. Consequently, combining some of these architectures leads to a more comprehensive and diverse set of extracted features (Alzubaidi et al., 2021).

In this study, a novel approach was used to generate Deep Dream images by employing the concept of stacking ensemble learning with multiple CNN variants, including Inception v3 (Lin et al., 2019), Xception (Jebur et al., 2023), VGG19 (et al., 2019), Inception-ResNet-V2 (Wang et al., 2021), and VGG16 (Al-Khazraji et al., 2022). The architectures in this study were stacked with different configurations ranging from three to five variants used simultaneously. In contrast with conventional Deep Dream generation methods, which rely on repeatedly feeding the input image and applying general processes, an alternative approach was adopted in this study. The images were generated by applying Deep Dream to each CNN variant, followed by the application of transformations and operations on the resulting images. Performance of the model was evaluated by computing overall average and loss across all variants.

The output yielded multiple Deep Dream images that mimic the hallucinations experienced by schizophrenia patients and addicted individuals. Similar to surrealist images, images generated in this study can also be considered as art since only fictitious shapes and figures were generated. The proposed model was evaluated by computing the loss quantifying the disparity between the generated and the target images. As such the novelty of the proposed work is presented as follow:

- As per the knowledge of the authors, the concept of stacking ensemble with multiple CNN variants is applied for Deep Dream for the first time.
- Multiple octaves were used for the implementation of the stacked Deep Dream model.
- CNN variants used in this study were carefully fine-tuned to outperform other hybrid models.

The reminder of this paper is organized as follows. Section two reviews the relevant literature. Section three shows theoretical background of the technologies used. Section four explains the methodology of this study. Experimental results and discussion are illustrated in Section five, and finally the conclusion is presented in Section six.

2. Related works

Yin et al. (Yin et al., 2020) presented DeepInversion, which involves both teacher and student logits. After training a neural network on a specified image, the teacher model produced unnormalized outputs. Simulations of teacher logits were generated from a separately trained model on student logits. The authors enhanced the appearance of images via the uniform distribution of the extracted features, thereby creating DeepInversion. The model was evaluated using the CIFAR-10 (The CIFAR 10 dataset, 2024) and ImageNet (Russakovsky et al., 2015) datasets. Two CNN variants, VGG-16 and ResNet, were used to construct DeepInversion model. However, the model lacks diversity, as using more variants would result in a stronger model and better dreamed images. Additionally, the model requires significant computational resources, and it is time-consuming.

Arthi et al. (Arthi et al., 2021) presented a security system that employs Deep Dream for extracting prominent features. Their architecture is an authentication system using biometrics. Deep Dream was used to generate dream-like images by amplifying particular layers in the CNN architecture, making the valuable features more prominent. The images generated by the Deep Dream model were then benchmarked with the biometric images in the dataset. Permission in the authentication system is granted or rejected based on the strength of match between the images generated by the model and images contained in the biometric image set and an alarm is activated to alert the system. Specific details of the CNN configuration in their study are unknown and the details of constructing the model or using a pretrained model were not clarified. The model was primarily constructed to function as a feature extractor for an authentication system, so the authors did not focus on Deep Dream as an image generator or on the quality of the generated images.

Al-Khazraji et al. (Al-Khazraji et al., 2023) presented a study combining Deep Dream and neural style transfer (NST) techniques. For Deep Dream, five CNN variants were used sequentially to generate dream images, which were then used as input for the NST method. When generating Deep Dream images, the model activated the upper layers, focusing on the more valuable features, and used the Adam optimizer during the training process. The stochastic gradient descent (SGD) optimizer was also tested, which demonstrated weaker performance compared to the Adam optimizer. This approach is time-consuming and requires significant memory to store the resulting images. Additionally, because NST is a highly complex method, it causes unacceptable delays and demands substantial computational resources.

Al-Khazraji et al. (Al-Khazraji et al., 2022) presented a Deep Dream method using the VGG-16 network. They examined the network and selected specific layers to activate. Initially, they activated the lower layers of VGG-16 by maximizing their activations, which amplified the features of the selected layers. They repeated this process with the upper layers of the network, maximizing the activations of these layers, thereby making other features more prominent. The authors concluded that the images resulting from activating the upper layers are more distinct and dream-like than those generated by activating the lower layers. Despite their considerable efforts, the study only employed the VGG-16 architecture, which is relatively simple compared to more complex architecture such as VGG-19, Inception, or Xception, or even a hybrid model combining multiple architectures. Consequently, the Deep Dream images generated using VGG-16 are less striking compared to those produced by more complex architectures.

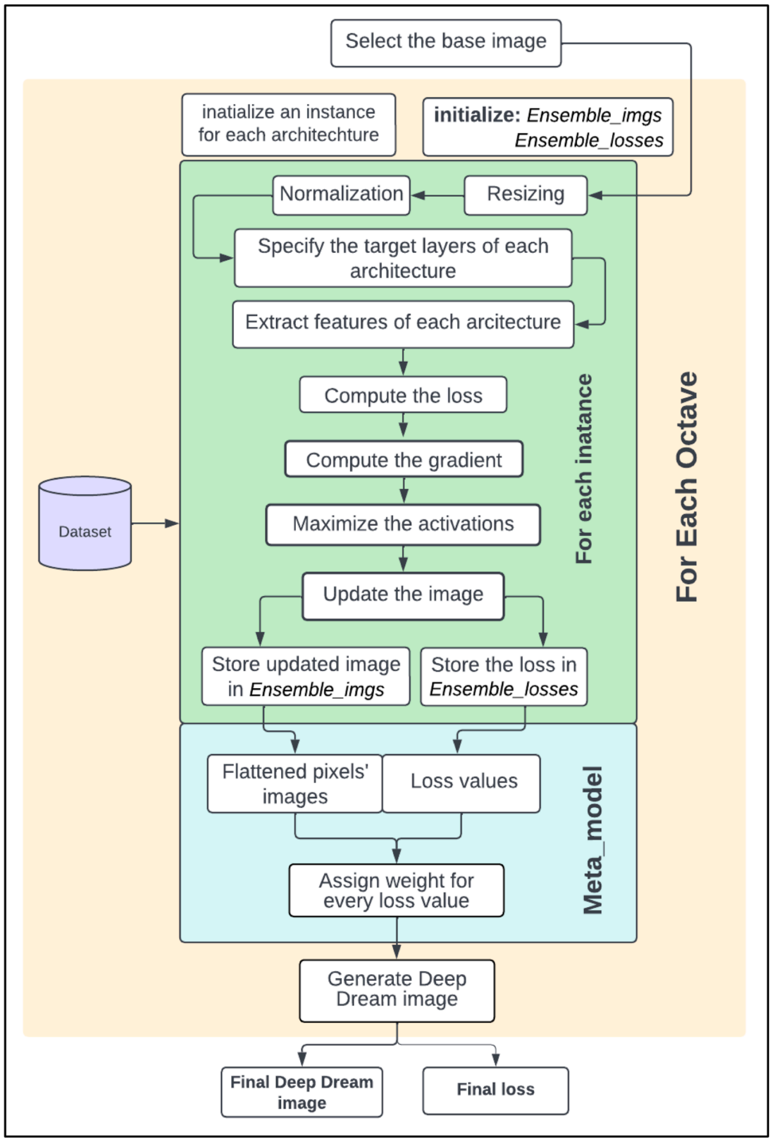

Fig. 1. Diagram of the stacking deep dream approach.

Sahu et al. (Sahu et al., 2023) presented a hybrid method that integrates features of machine learning and deep learning. The authors used the Inception v3 network with the original Deep Dream algorithm without any modifications. They collected input images from farms, aiming to extract features of disease lesions on plant leaves. The Deep Dream algorithm amplifies these features by maximizing the activations through the application of gradient ascent on specified layers of the Inception network, making the features more distinct and prominent. Generally, this method focuses solely on feature extraction rather than producing Deep Dream images.

Al-Khazraji et al. (Al-Khazraji et al., 2024) presented a Deep Dream method for generating dreamed images using two CNN variants, Inception-ResNet-V2 and Inception-v3, each working independently. They implemented their method by targeting the lower layers, then targeting the upper layers. This approach generated four types of Deep Dream images: the first two resulted from activating the lower layers in each network, while the last two were produced by activating the upper layers. However, this method produced weaker dreamed images, as each variant was used independently, meaning the model only focused on the part of the input image it could see, unlike the hybrid methods, which target the input image from multiple perspectives.

From the above studies, two gaps can be identified:

- Some of these studies as in Al-Khazraji et al., (2022); Yin et al. (2020), and (Al-Khazraji et al., 2024), lack diversity in the CNN architectures. They produced simple and not attractive images due to the use of only one or two architectures, and each model is utilized independently without any combination or hybridization between models.

Other models, as indicated in Arthi et al., (2021) and (Sahu et al., 2023), use Deep Dream to maximize specific features. These models are used as feature extractors rather than dream generators.

To address these gaps, we are proposing the following improvements:

- This study combines five CNN architectures in a novel way, rather than using one or two architectures independently.
- The main task of this study is to generate Deep Dream images, not feature extraction or selection.
- The study employs the concept of stacking ensemble learning to combine all five CNN architectures effectively.

Table 1
The values of hyperparameters.

No.	Hyperparameter	Value
1	Learning rate	0.01
2	Number of iterations	1000
3	Number of octaves	3
4	Octave's scaling factor	1.3
5	Optimizer	Adam

3. Methodology

The proposed study was carried out by including stacking ensemble learning approach using Google Colab, with Keras and TensorFlow libraries. Fig. 1 illustrates the schematic of the proposed model. The first step in this process is preprocessing, which includes resizing and normalization. During resizing, the base image was adjusted to the required size which is either 224 × 224 for VGG-16 and VGG-19, or 299 × 299 for the Inception, Xception, and Inception-ResNet-V2, this can be achieved by changing the dimensions (width and height) of an image while maintaining its aspect ratio or distorting it to fit the desired dimensions.

The new size of the image must maintain the aspect ratio, ensuring that the proportion of the image's width to height is preserved. This is achieved by applying

$$x'/y' = x/y$$

where x' and y' represent the new dimensions of the images, while x and y are the original dimensions. Normalization is then applied to scale the values of data to be in the range of $0 - 1$. Normalization is achieved by applying $x_n = (x_r - x_{min} / x_{max} - x_{min})$ (Hamori et al., 2018; Yin et al., 2017) where x_r denotes the intensity value of a pixel, x_n is the normalized intensity value and x_{max} and x_{min} denote the maximum and the minimum intensity values of an image, respectively.

When implementing the CNN variants, features were detected based on which layers are activated, which means amplified specific features or patterns in those layers. In this model, only the deep layers were unfrozen, as they contain the most important details.

The hyperparameters used for each of the CNN variants are shown in Table 1.

The Deep Dream instances were created, and each one was associated with a different CNN variant, where the stacking ensemble of Deep Dream instances was represented by the Stacking Ensemble class, in which a list of Deep Dream instances was accepted as an input. The resulting Deep Dream image from the first instance was passed to the second instance and the process of Deep Dream was applied to it and so on, and each instance passed the resulting image to the next one until all the instances are completed. Therefore, the ensemble was used to produce the Deep Dream image by going through each deep dream instance iteratively, gathering the generated images and losses, and determining the weights for each model using the results of a meta-model. Each instance of a Deep Dream that makes up the ensemble produces an image and calculates a loss value. Two lists were initialized which are *Ensemble_losses* and *Ensemble_imgs. Ensemble_losses* were reshaped into a column vector with the shape (−1, 1), and Ensemble_imgs was reshaped into a 2D array to prepare the input for *meta_model*. Each row in the reshaped Ensemble_imgs represents a deep dream instance, flattening the picture pixels into a single dimension. A 2D array named meta_model_input is created by horizontally stacking the revised ensemble_losses and ensemble_imgs. The loss is computed by applying Eq. (1). Where generally, the loss quantifies the disparity between the generated image and the target image. Throughout the training process, the goal is to minimize the loss in order to enhance the similarity between the generated image and the desired image. Many metrics can be used to measure the loss function such as Mean Squared Error (MSE), Euclidian Distance, Eq. (1) is used to compute the loss function between two

Algorithm 1
The proposed stacking deep dream system.

Input
- Base image, Learning rate μ, Number of steps N
- Counter values: $P = 1,\ i = 1$
- Number of instances, F
- Number of octaves, *octaves*
- Octave_scale, *SCALE*

Output
- Final Deep Dream image.
- Final loss.

Begin
1. Initialize lists:
   1.%2. *Ensemble_imgs* /* to store the generated dream images.
   2.%2. *Ensemble_losses* /* to store the corresponding loss values of those images.
2. Input the base image, *img*.
3. Create an instance for each Deep Dream model, each one corresponds to a CNN architecture.
4. Resize the base image to the required size.
5. Normalize the resized base image.
6. Calculate the scaling factor, where $S = \text{SCALE}^{K}$
7. For $i \le octaves$ Do:
   1.%2. While $P \le F$ Do:
      1.%2.%3. For $i = 1 to\ N$ Do:
         I. Take (img) as the base image.
         II. Scale the image by applying Eq. (5).
         III. Select the target layers from the network.
         IV. Extract the features by applying CNN roles.
         V. Compute the loss of the image by applying Eq. (2).
         VI. Calculate the gradient ascent of each pixel in the layers by applying Eq. (3).
         VII. Maximize the activations in each layer.
         VIII. Update the image in the orientation of the gradient by applying Eq. (4).
         IX. Store *DeepDream_image* in *Ensemble_imgs*.
         X. Store *loss* in *Ensemble_losses*.
         XI. Display (DeepDream_*image, loss*)
      2.%2.%3. $P = P + 1$

End While.
1. Flatten all images in the *Ensemble_imgs* in the *meta_model*.
2. Put the loss values in the *meta_model*.
3. In the *meta_model*, assign a weight for every loss value.
4. Generate the Deep Dream image depending on the weights of the losses.
5. Display the final Deep Dream image and the final loss.

End.

images based on MSE (Gao & Chen, 2022; Nielsen, 2015).

$$C(w, b) = 1 \Big/ 2n \sum_x \| y(x) - a \|^2 \tag{1}$$

Where $C(w, b)$ represents the loss, (w, b) refers to the weight and bias, n is the number of training examples, a represents the magnitude of the error, and $y(x)$ represents the real target for the input x which produces the predicted output.

But in the case of Deep Dream, there is no target image, where the input base image is modified in each iteration until reaching the Deep Dream image, thus the loss here is maximized each time and the pixels are adjusted each iteration depending on increasing the loss and the gradient ascent direction. Here, the loss in Deep Dream is calculated based on Eq. (2) (Gao & Chen, 2022; Nielsen, 2015).

$$Loss(X_i, X_{base}) = \frac{1}{n} \ \| F(X_i) - F(X_{base}) \|^2 \tag{2}$$

Where $Loss(X_i, X_{base})$ indicates the value of loss between the feature activations of the generated Deep Dream image at iteration i and the base image, $F(X_i)$ represents the feature activations obtained by passing the Deep Dream image at iteration i through the CNN architecture and $F(X_{base})$ represents the feature activations obtained by passing the base image through the same CNN architecture.

The gradient ascent can be computed based on the activations of the neurons in each layer, as shown in Eq. (3) (Suzuki, et al., 2017).

$$gradientascent_i = \nabla_i Loss(x) \tag{3}$$

Where $\nabla_i Loss(x)$represents the gradient of the loss $Loss$ with respect

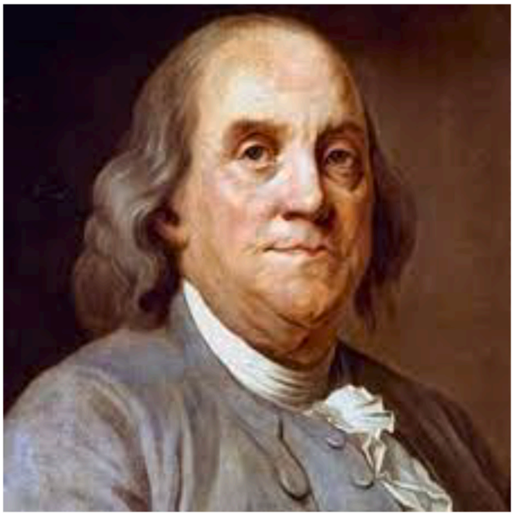

Fig. 2. The base image.

to the base image x; which will be the updated image in the later stages and i refers to the number of iterations.

Thus, from the fact that says, the weights in Deep Dream are not updated; instead, the base image is updated, Eq. (4) is used to update the base image at each iteration (Suzuki et al., 2017).

$$x_{new} = x_{old} + \mu . \ gradientascent \tag{4}$$

Where x_{old} is the image at this moment, while x_{new} is the updated image. μ is the learning rate. Thus, μ controls the process of updating the base image, achieving an equilibrium between optimal and large modifications of the image.

This process is repeated for a specified number of octaves that implement the Deep Dream operation with different sizes based on the scaling factor. Where image are resized according to the octaves as in Eq. (5) (Hsieh et al., 2021).

$$x[K] = resize \ (I[K-1], \ scaling = S) \tag{5}$$

Where $x[K]$ is the modified image after applying the octave. S is the scaling factor in each octave, SCALE is the octave_scale value which is less than one in the case of downsizing and greater than one in the upsizing, and K represents the number of octaves.

The outputs of the Deep Dream instances are combined using *meta_model* to give weights to each loss of instance, where the weights are given based on the values of the losses; the small loss value has a small weight, and the high loss value has a high weight. Thus, five final loss values are obtained; one for each instance, which has a range between 0 and 1, and their sum must be 1.

Algorithm 1 shows the proposed model steps in detail.

4. Experimental results and discussion

Various experiments were conducted to test the proposed model. In this case the base image of the former American president Franklin was used as shown in Fig. 2.

Then two preprocessing steps are utilized including resizing and normalization. In which the resized and normalized image is used as input to the proposed Deep Dream model. Fig. 3 shows the normalized image.

This image is then input to the stacking ensemble Deep Dream model which consists of five CNN architectures integrated together: VGG-16, Inception v3, VGG-19, Inceptio-ResNet-V2, and Xception. The result is three Deep Dream images, each one represents an octave. Fig. 4 shows the resulting Deep Dream image when applying the model with the first octave. While Figs. 5 and 6 show the Deep Dream images with the second and third octaves, respectively.

The final Deep Dream image is generated from all three octave images as illustrated in Fig. 7.

The loss of each one of the generated Deep Dream images is listed in Table 2.

This model produced Deep Dream images with intricate details, where activations occur in the top three layers, which are rich in features. This leads to the appearance of distinctive patterns and figures in the image. The depth and repeated convolution operations in VGG-19 cause specific patterns to be repeatedly maximized in these top layers. Additionally, the extreme depth of the Xception and Inception v3 networks allows them to target the image from different perspectives, while VGG-16 focuses on patterns with less detail compared to the other networks. Finally, Inception-ResNet-V2 extracts and maximizes features in a way that differs from all other networks. The activation function (ReLU in this case) has a significant impact on the resulting Deep Dream image. From the loss, it is evident that at the third octave, the complexity of the resulting image increases. This is due to the repeated feeding of the image into the model, where the image first passes through the model with the first octave, resulting in limited loss. The image is then passed through the model again with the second octave, where the loss increases significantly. When the image enters the model a third time with the third octave, the loss continues to rise. Thus, as the number of octaves increases, the loss escalates substantially. We also implemented the stacking ensemble Deep Dream model by adding EfficientNet and

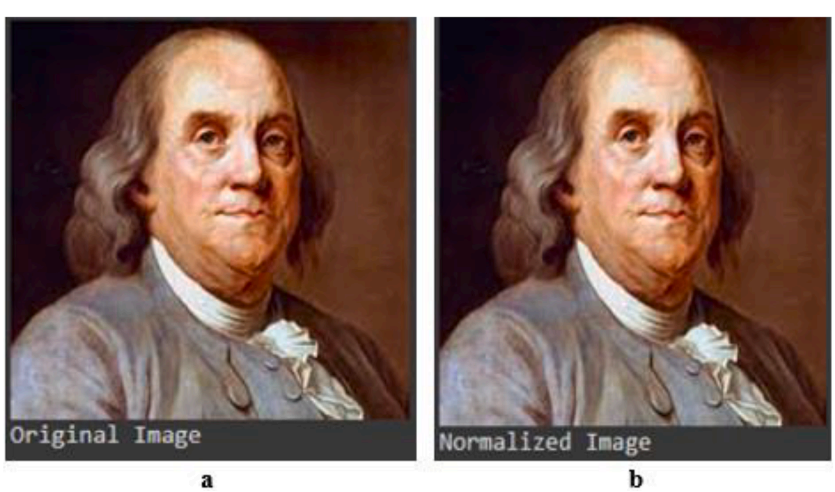

Fig. 3. a) The Original base image, b) Image after applying the normalization process.

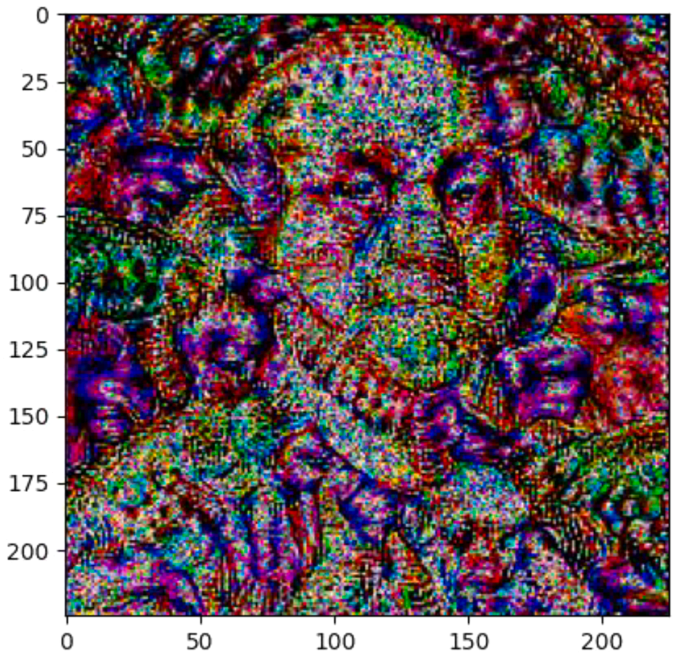

Fig. 4. The resulting deep dream image with the first octave.

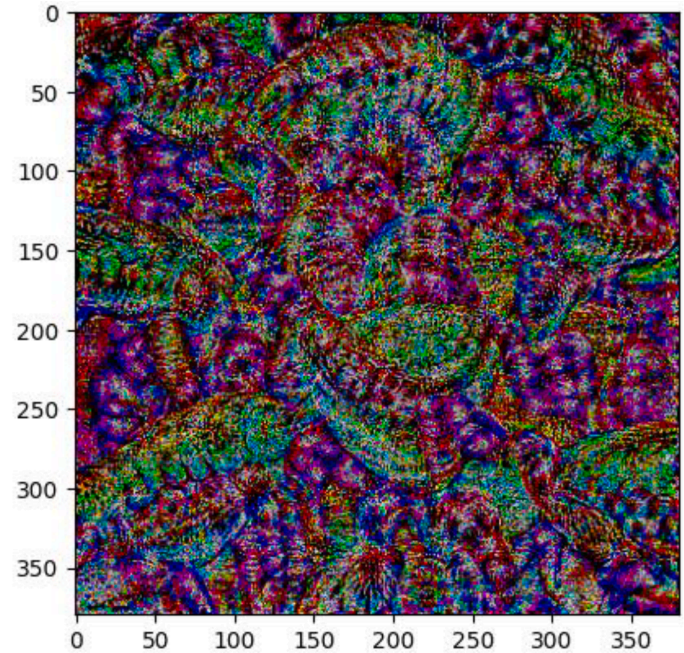

Fig. 6. The resulting deep dream image with the third octave.

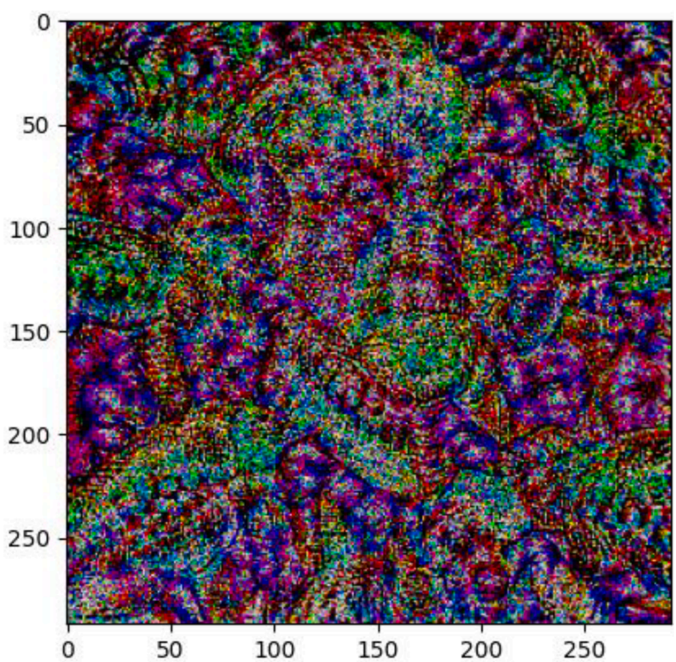

Fig. 5. The resulting deep dream image with the second octave.

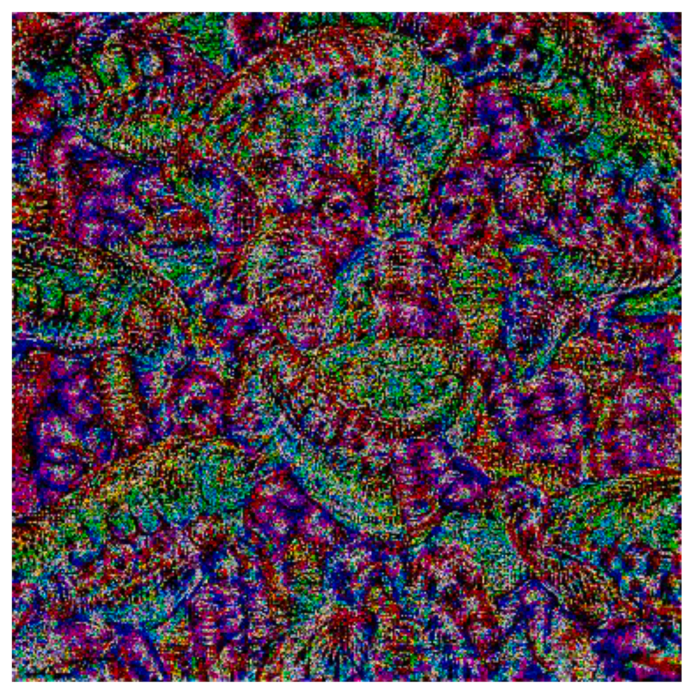

Fig. 7. The final deep dream image from the proposed stacking deep dream model.

removing VGG-16 and VGG-19. The resulting Deep Dream image is shown in Fig. 8, with a loss value of 4.0939. It is evident that the Deep Dream image in Fig. 8 did not maintain the details, as EfficientNet is a lightweight network and does not have a complex structure compared to VGG-19, which focuses on large patterns in the base image. The absence of VGG-16, which preserves small details, is also a factor. This is the reason the image in Fig. 8 appears distorted and does not represent a dream-like image. Additionally, the loss is very low because the patterns were not amplified, resulting in low detail, such as lines and edges. The stacking Deep Dream model still has some limitations. It requires high computational resources, as most deep learning techniques that deal with images demand significant memory space and strong processing

Table 2
The loss values of the stacking ensemble deep dream model.

Image	Average loss value
The first octave image	9.2122
The second octave image	52.8167
The third octave image	80.7174
The average loss of the final image	47.5821

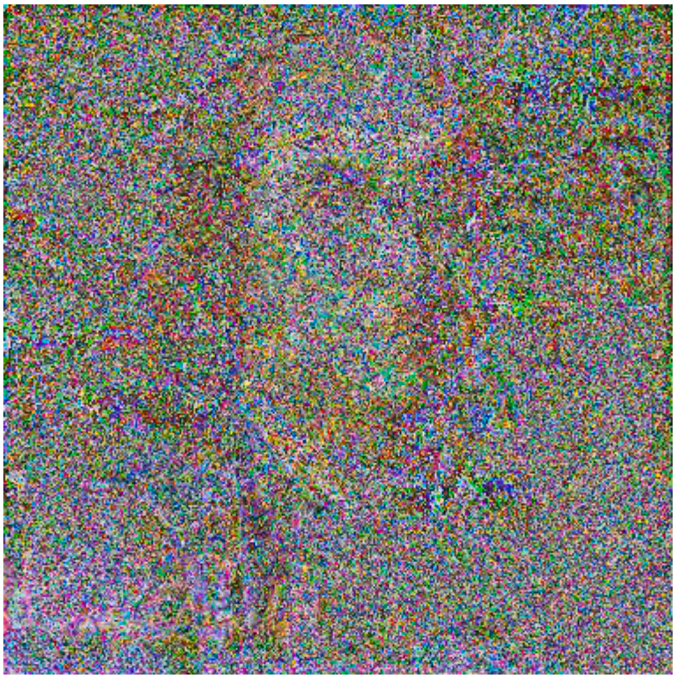

Fig. 8. The resulting deep dream image based on stacking ensemble deep dream model using inception v3, inception-ResNet-V2, Xception, and EfficientNet.

power. Additionally, it is a time-consuming method since it continues execution during each octave, with each variant implemented and saving numerous images—one image per 50 iterations—along with the loss of each image. At the end of the process, the average loss is calculated, and the final Deep Dream image is produced.

5. Conclusion

Deep Dream technology has recently emerged as one of the most promising computer vision techniques. In this study, a novel method was presented for generating Deep Dream images by employing the concept of stacking ensemble learning with multiple CNN architectures. This approach generated dream-like images that mimic the hallucinations seen by addicts and people who suffer from schizophrenia, and the generated images were considered as artworks due to their surrealistic appearance. Five CNN architectures—VGG-16, Inception v3, VGG-19, Inception-ResNet-V2, and Xception were integrated based on the stacking ensemble concept. The base image was processed by the Deep Dream model, implemented using one of these networks, and the generated image was stored. The next image was then stacked above the previous one, and this process continued until the completion of image generation process. The procedure was then repeated with the next CNN variant, following the same steps as the first variant, until all variants were used, and three octaves were completed. The average loss and final Deep Dream image were then extracted. The loss value served as a metric to assess how much the dreamed image differed from the base image while still retaining prominent features that allow for the image's recognition. The proposed method is limited by the high computational resources and time required. The future extension of this study is aimed at reducing the complexity in terms of both space and time. Additionally, development of the Deep Dream model will be attempted using other suitable ensemble learning approaches.

CRediT authorship contribution statement

Lafta Alkhazraji: Conceptualization, Methodology, Writing – original draft. **Ayad R. Abbas:** Data curation, Formal analysis, Writing – review & editing. **Abeer S. Jamil:** Software, Validation, Visualization. **Zahraa Saddi Kadhim:** Resources, Supervision. **Wissam Alkhazraji:** Project administration, Writing – review & editing. **Sabah Abdulazeez Jebur:** Investigation, Data curation. **Bassam Noori Shaker:** Writing – review & editing, Validation. **Mohammed Abdallazez Mohammed:** Funding acquisition, Supervision. **Mohanad A. Mohammed:** Visualization, Investigation. **Basim Mohammed Al-Araji:** Data curation, Resources. **Abdulkareem Z. Mohmmed:** Investigation, Resources. **Wasiq Khan:** Methodology, Writing – review & editing. **Bilal Khan:** Methodology, Writing – review & editing. **Abir Jaafar Hussain:** Methodology, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

"Schizophrenia world human organization (WHO)." Accessed: Aug. 23, (2024). [Online]. Available: Https://www.who.int/news-room/fact-sheets/detail/schizophrenia?gad_source=1&gclid=Cj0KCQjww5u2BhDeARIsALBuLnPVecYB35PyJIb_cWC-rZnk2dcEX–Y-dj-wqBPHeRIeOyKhtL-ANQaAtCwEALw_wcB.

"World human organization (WHO) Drugs (psychoactive)." Accessed: Aug. 23, (2024). [Online]. Available: Https://www.who.int/health-topics/drugs-psychoactive#tab=tab_2.

"The CIFAR 10 dataset." Accessed: Sep. 15, (2024). [Online]. Available: Https://www.kaggle.com/datasets/ekaakurniawan/the-cifar10-dataset.

Ali, L. R., Jebur, S. A., Jahefer, M. M., & Shaker, B. N. (2022). Employing transfer learning for diagnosing COVID-19 disease. *International Journal Of Online And Biomedical Engineering, 18*(15), 31–42. https://doi.org/10.3991/ijoe.v18i15.35761

Ali, L. R., Shaker, B. N., & Jebur, S. A. (2023). An extensive study of sentiment analysis: A survey. In *AIP Conference Proceedings 2591*, Article 030022. https://doi.org/10.1063/5.0119604

Al-Khazraji, L. R., Abbas, A. R., & Jamil, A. S. (2022). The effect of changing targeted layers of the deep dream technique using VGG-16 Model. *International Journal Of Online And Biomedical Engineering., 19*(3), 34–47.

Al-Khazraji, L. R., Abbas, A. R., & Jamil, A. S. (2023). A systematic review of deep dream. *The Iraqi Journal of Computers,Communications,Control and Systems Engineering, 23*(2), 192–209. https://doi.org/10.33103/uot.ijccce.23.2.15

Al-Khazraji, L. R., Abbas, A. R., & Jamil, A. S. (2024). Generating various deep dream images through maximizing the loss function of particular layers using inception-v3 and inception-ResNet-V2 models. *Iraqi J. Sci., 65*(6), 3468–3483. https://doi.org/10.24996/ijs.2024.65.6.39

Al-Khazraji, L. R., Abbas, A. R., Jamil, A. S., & Hussain, A. J. (2023). A hybrid artistic model using deepy-dream model and multiple convolutional neural networks architectures. *IEEE Access : Practical Innovations, Open Solutions, 11*, 101443–101459. https://doi.org/10.1109/ACCESS.2023.3309419

Alzubaidi, L., et al. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data, 8*(1), 1–74. https://doi.org/10.1186/s40537-021-00444-8

Apollinaire, G., & Manifesto,.S. (2023). "Surrealism art movement: history, characteristics, artwork,"[Online]. Available: Https://www.artchive.com/art-movements/surrealism/.

Arthi, R., Kishan, A. R., Abraham, A., & Sattenapalli, A. (2021). Centralized intelligent authentication system using deep learning with deep dream image algorithm. In *Advances in power systems and energy management: Select proceedings of ETAEERE 2020* (pp. 169–178). Springer. https://doi.org/10.1007/978-981-15-7504-4_18.

Gao, Z., & Chen, J. (2022). Maritime infrared image super-resolution using cascaded residual network and novel evaluation metric. *IEEE Access : Practical Innovations, Open Solutions, 10*, 17760–17767. https://doi.org/10.1109/ACCESS.2022.3147493

Hamori, S., Kawai, M., Kume, T., Murakami, Y., & Watanabe, C. (2018). Ensemble learning or deep learning? Application to default risk analysis. *Journal of Risk and Financial Management, 11*(1), 12. https://doi.org/10.3390/jrfm11010012

Hazar, M. J., Shaker, B. N., Ali, L. R., & Alzaidi, E. R. (2020). Using received strength signal indication for indoor mobile loscalization based on machine learning technique. *Webology, 17*(1), 30–42. https://doi.org/10.14704/WEB/V17I1/A206

Hsieh, K. H., Chen, E. C., & Lee, C. R. (2021). FOCM: Faster octave convolution using mix-scaling. In *International conference on technologies and applications of artificial intelligence, TAAI . IEEE* (pp. 31–36). https://doi.org/10.1109/TAAI54685.2021.00015

Jebur, S. A., Hussein, K. A., Hoomod, H. K., & Alzubaidi, L. (2023). Novel deep feature fusion framework for multi-scenario violence detection. *Computers, 12*(9), 175. https://doi.org/10.3390/computers12090175

Lin, C., Li, L., Luo, W., Wang, K. C. P., & Guo, J. (2019). Transfer learning-based traffic sign recognition using inception-v3 model. *Periodica Polytechnica Transportation Engineering., 47*(3), 242–250. https://doi.org/10.3311/PPtr.11480

Nielsen, M. A. (2015). *Neural networks and deep learning, 25*. San Francisco, CA, USA: Determination press. https://doi.org/10.1108/978-1-83909-694-520211010

Rastelli, C., Greco, A., Kenett, Y. N., Finocchiaro, C., & De Pisapia, N. (2022). *Simulated visual hallucinations in virtual reality enhance cognitive flexibility*. Nature Publishing Group UK. https://doi.org/10.1038/s41598-022-08047-w

Russakovsky, O., et al. (2015). ImageNet large scale visual recognition challenge. *International Journal Of Computer Vision, 115*(3), 211–252. https://doi.org/10.1007/s11263-015-0816-y

Sahu, P., Chug, A., Singh, A. P., & Singh, D. (2023). Classification of crop leaf diseases using image to image translation with deep-dream. *Multimedia Tools and Applications*, 1–35. https://doi.org/10.1007/s11042-023-14994-x

Suzuki, K., Roseboom, W., Schwartzman, D. J., & Seth, A. K. (2017). *A deep-dream virtual reality platform for studying altered perceptual phenomenology*. US: Springer. https://doi.org/10.1038/s41598-017-16316-2

Wang, J., He, X., Faming, S., Lu, G., Cong, H., & Jiang, Q. (2021). A real-time bridge crack detection method based on an improved inception-resnet-v2 structure. *IEEE access : Practical Innovations, Open Solutions, 9*, 93209–93223. https://doi.org/10.1109/ACCESS.2021.3093210

Wen, L., Li, X., Li, X., & Gao, L. (2019). A new transfer learning based on VGG-19 network for fault diagnosis. In *Proceedings of the 2019 IEEE 23rd international conference on computer supported cooperative work in design* (pp. 205–209). https://doi.org/10.1109/CSCWD.2019.8791884

Yin, H., et al. (2020). Dreaming to distill: Data-free knowledge transfer via deepinversion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 8715–8724). https://doi.org/10.1109/CVPR42600.2020.00874

Yin, Z., Wan, B., Yuan, F., Xia, X., & Shi, J. (2017). A deep normalization and convolutional neural network for image smoke detection. *IEEE Access : Practical Innovations, Open Solutions, 5*, 18429–18438. https://doi.org/10.1109/ACCESS.2017.2747399