

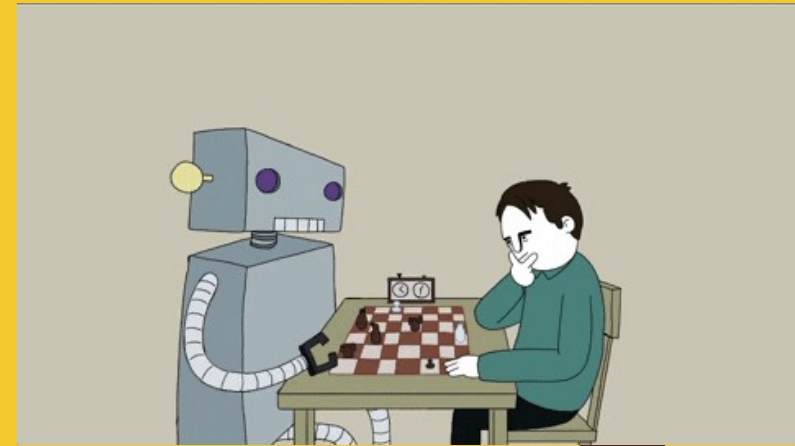
CSE4006

DEEP LEARNING

Dr K G Suma

Associate Professor

School of Computer Science and Engineering



Module No. 3

Convolution Neural Networks

7 Hours

- Convolutional networks optimization
- Loss functions in classifiers
- Convolution layers
- Max pool layers
- VGG
- Google Net
- ResNet
- Dropout
- Normalization
- Rules update
- Data augmentation
- Transfer learning
- Analysis of pre trained models

Data Augmentation

- Data augmentation is a powerful technique that has gained significant traction in the field of Deep learning, particularly in areas where data scarcity poses a challenge.
- By artificially increasing the size and diversity of the training dataset, data augmentation enables Deep learning models to learn more robust and generalizable representations, leading to improved performance and accuracy.

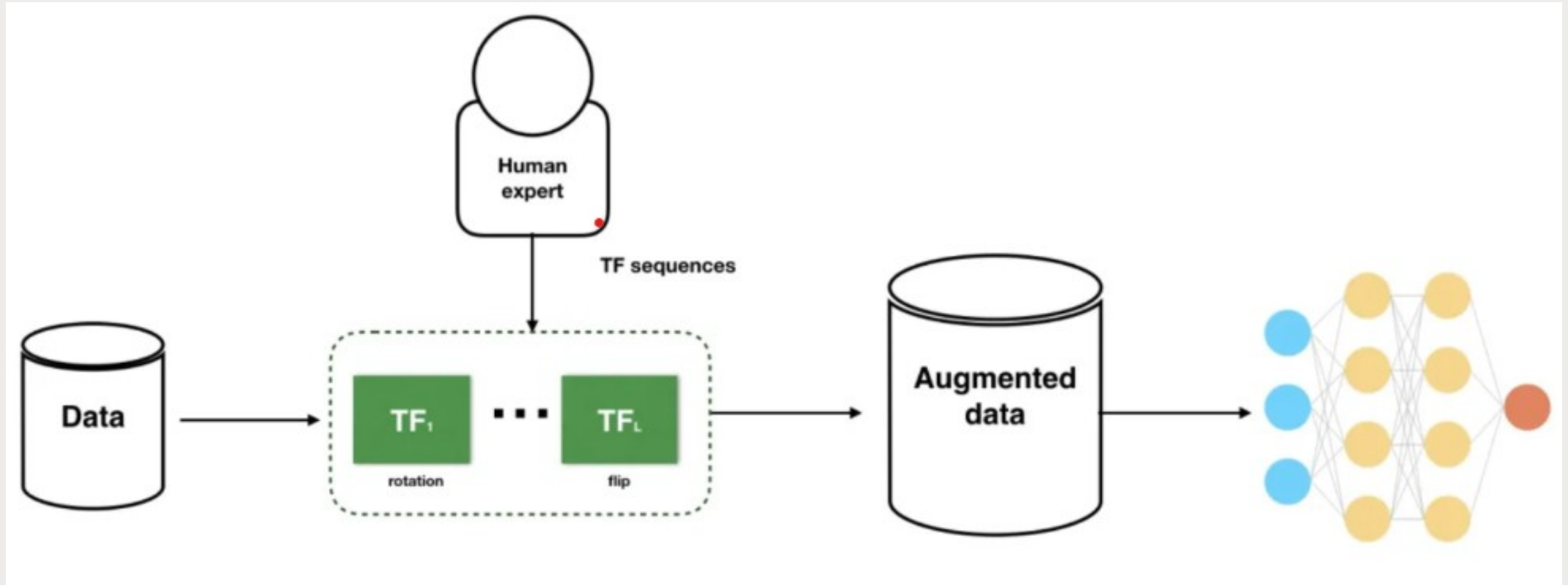
Data Augmentation

- Data augmentation is a set of techniques used to artificially expand the size of a training dataset by creating modified versions of existing data. These modifications can range from simple transformations, such as flipping or rotating images, to more complex techniques, like generating synthetic data using Deep learning models.
- The primary goal of Data augmentation is to increase the diversity and variability of the training data, thereby exposing the Deep learning model to a wider range of scenarios and reducing the risk of overfitting.

(Overfitting occurs when a model learns the noise or irrelevant patterns in the training data instead of capturing the underlying patterns, leading to poor generalization performance on unseen data)

- Data augmentation can be applied to various types of data, including images, text, audio, and time-series data. However, it is particularly prevalent in the field of computer vision, where image data is abundant, and data augmentation techniques can be readily applied.

Data Augmentation

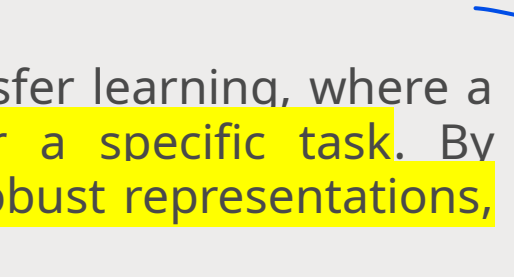


Why is Data Augmentation Important?

Data augmentation has become increasingly important in machine learning for several reasons:

- **Limited Data Availability:** In many real-world applications, obtaining large, high-quality datasets can be challenging, time-consuming, and expensive. Data augmentation provides a cost-effective way to increase the size and diversity of the training data, enabling the development of more accurate and robust models.
- **Overfitting Prevention:** By introducing variations in the training data, data augmentation helps prevent overfitting, a common issue in Deep learning where models perform well on the training data but fail to generalize to unseen data. Augmented data exposes the model to a broader range of scenarios, improving its ability to generalize.
- **Improved Model Performance:** Data augmentation can significantly improve the performance of machine learning models, particularly in tasks such as image recognition, object detection, and natural language processing. Augmented data provides additional examples for the model to learn from, leading to better feature extraction and increased accuracy.

Why is Data Augmentation Important?

- **Class Imbalance Mitigation:** In many classification problems, the distribution of classes in the training data can be imbalanced, with some classes being underrepresented. Data augmentation can be used to generate additional samples for the minority classes, mitigating the impact of class imbalance and improving the model's ability to correctly classify underrepresented classes.
 - **Transfer Learning:** Data augmentation can also benefit transfer learning, where a pre-trained model is fine-tuned on a smaller dataset for a specific task. By augmenting the target dataset, the model can learn more robust representations, improving its performance on the new task.
- 

Benefits of Data Augmentation

- **Increases Dataset Size:** Data augmentation allows expanding dataset size artificially when collecting more real data is costly or impractical.
- **Reduces Overfitting:** By introducing more varied data points, it helps regularize models and prevent overfitting to the training data.
- **Improves Accuracy:** Models trained on augmented datasets tend to achieve higher prediction accuracy by learning more robust representations.
- **Cost Saving:** It reduces expensive costs associated with data collection, cleaning and labeling by reusing existing data.
- **Handles Class Imbalance:** Over/undersampling minority classes using augmentation helps handle skewed class distributions.
- **Robustness:** Augmentations can introduce natural variations that increase the model's ability to generalize well.
- **Data Privacy:** Synthetic data can enable model training without compromising real data privacy and security.

In essence, data augmentation provides an efficient way to amplify and enrich training datasets, leading to more accurate, generalized, and cost-effective machine learning models across domains like computer vision, natural language processing, and speech recognition.

Limitations of Data Augmentation

While data augmentation is very useful, it also has some limitations:

- The augmented data inherits biases present in the original dataset
- Advanced quality checks are needed to ensure augmented data is realistic
- Research and specialized techniques are required for complex augmentation like generating high-resolution images
- Finding the right augmentation approach for a given problem can be challenging

Despite these limitations, data augmentation remains a powerful tool that can significantly boost machine learning performance when used appropriately.

Data Augmentation Techniques

Data augmentation techniques can be broadly categorized into two main approaches:

- Traditional transformations
- Deep learning-based methods

Traditional Transformations

- Traditional data augmentation techniques involve applying predefined transformations to the existing data. These transformations can be geometric, color-based, or a combination of both.

1. Geometric Transformations:

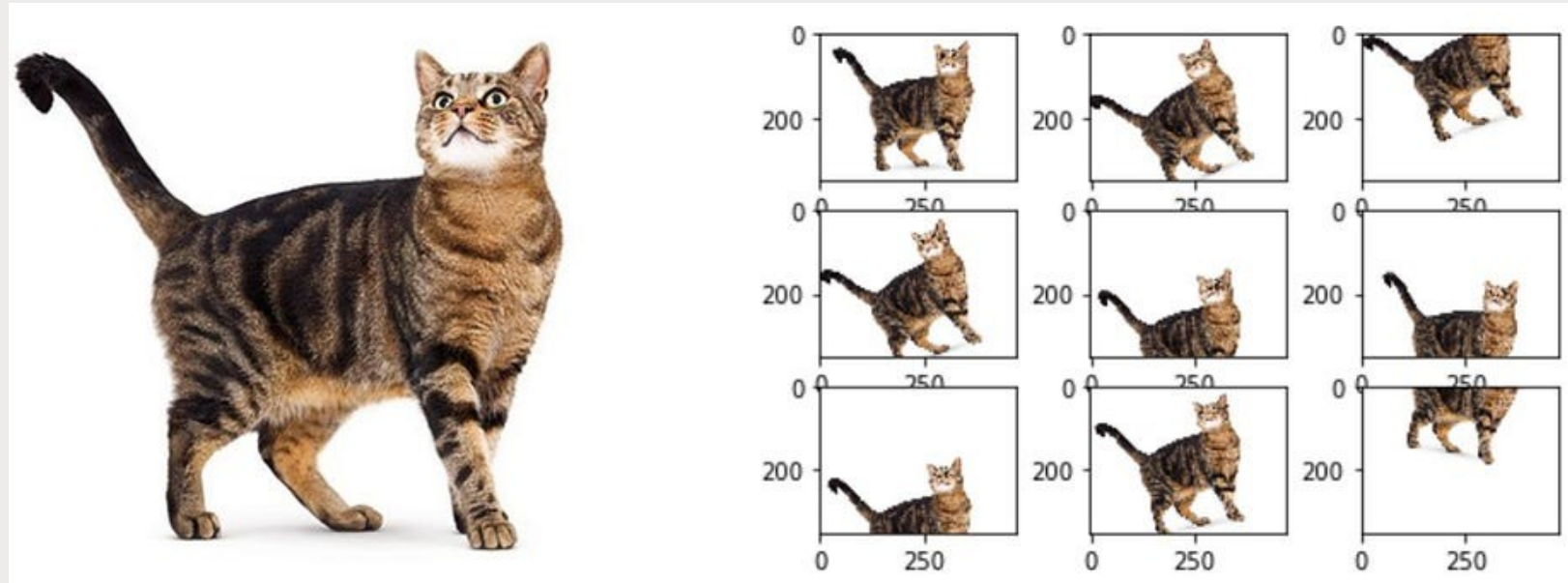
- **Flipping:** Horizontally or vertically flipping images.
- **Rotation:** Rotating images by a certain angle.
- **Cropping:** Extracting a region of interest from an image.
- **Scaling:** Resizing images to different scales.
- **Translation:** Shifting images horizontally or vertically (Shifting the image along the x or y axis).
- **Perspective Transformations:** Applying perspective distortions to images.
- **Shearing:** Slanting the shape of the image

2. Color-based Transformations:

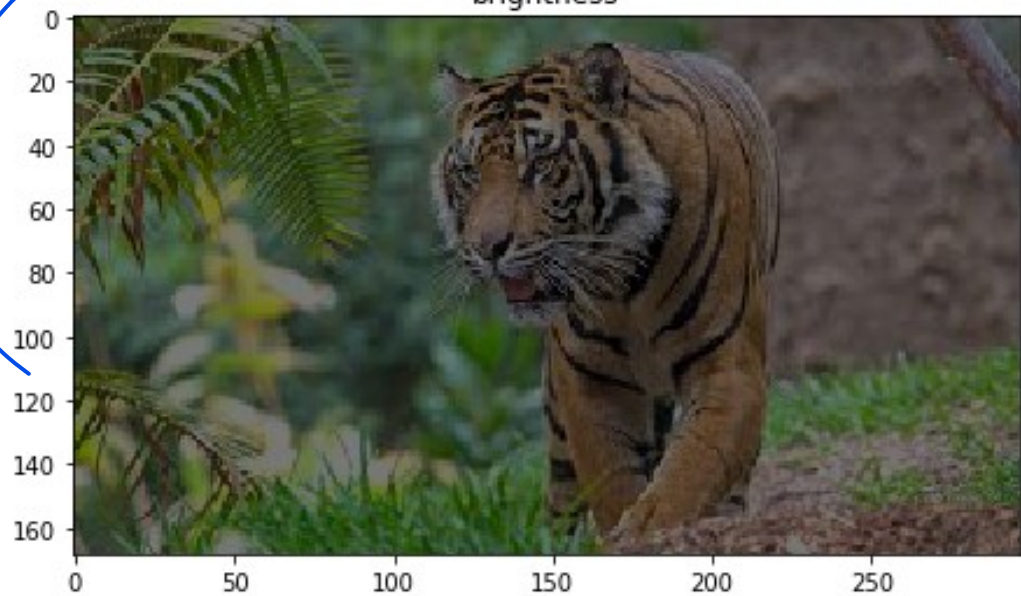
- **Brightness Adjustments:** Increasing or decreasing the brightness of images.
- **Contrast Adjustments:** Modifying the contrast levels of images.
- **Saturation Adjustment:** Modifying the intensity of colors in the image.
- **Hue Adjustment:** Shifting the colors by changing the hue.
- **Color Jittering:** Randomly adjusting the brightness, contrast, saturation, and hue of images.
- **Gray scaling:** Converting color images to grayscale.



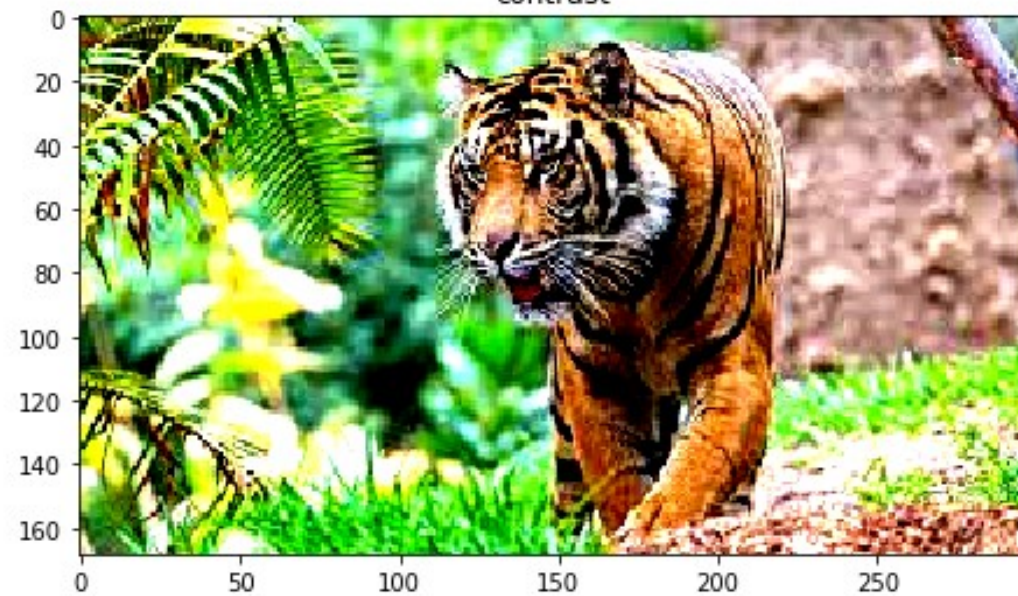
- Hue – Basic color
- Saturation - Intensity of the color
- Brightness – Intensity/ Quantity of the Light



brightness



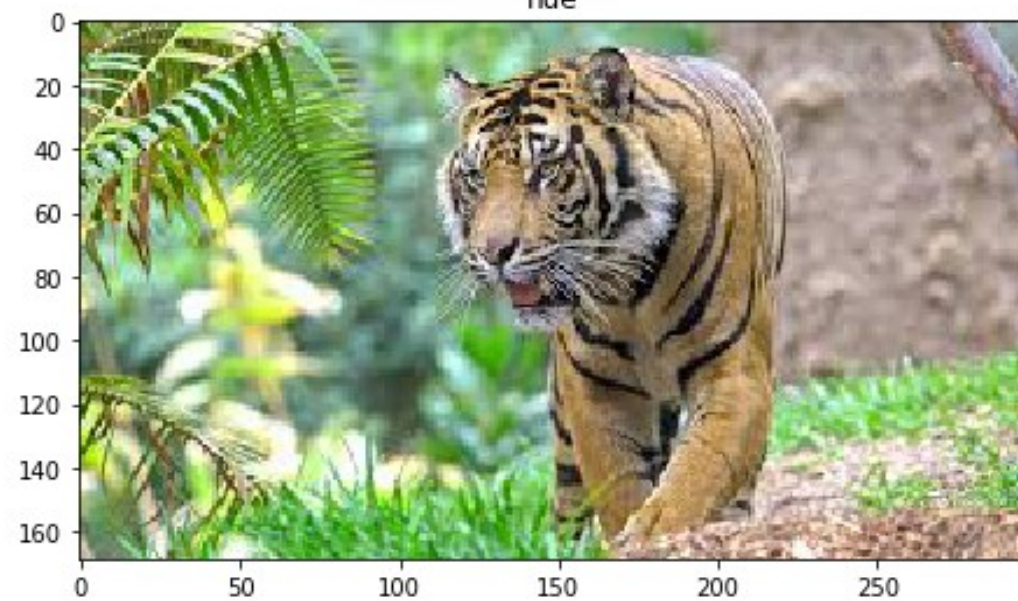
contrast



saturation



hue



Traditional Transformations

3. Noise Injection:

- **Gaussian Noise:** Adding random Gaussian noise to images or audio signals.
- **Salt-and-Pepper Noise:** Introducing random white and black pixels in images.

4. Kernel Filters: Kernel filters apply convolutional operations to enhance or suppress specific features in the image.

- **Blurring:** Applying Gaussian blur to smooth the image.
- **Sharpening:** Enhancing the edges to make the image sharper.
- **Edge Detection:** Highlighting the edges in the image using filters like Sobel or Laplacian.

5. Random Erasing

- Random erasing involves randomly masking out a rectangular region of the image. This helps the model become invariant to occlusions and improves its ability to handle missing parts of objects.

6. Combining Augmentations

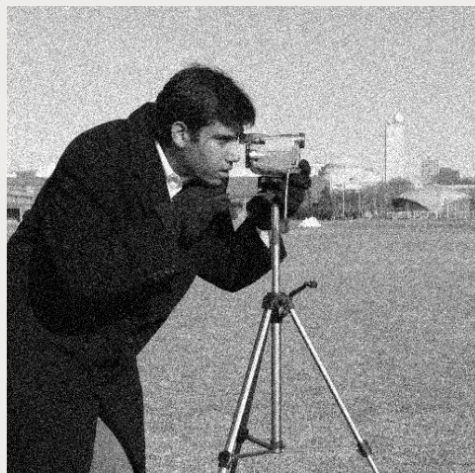
- Often, multiple augmentation techniques are combined to create more varied training data. For example, an image might be rotated, flipped, and then have its brightness adjusted in a single augmentation pipeline.

These simple transformations are easy to apply and can significantly increase the diversity of image, audio, and text data.

Original image



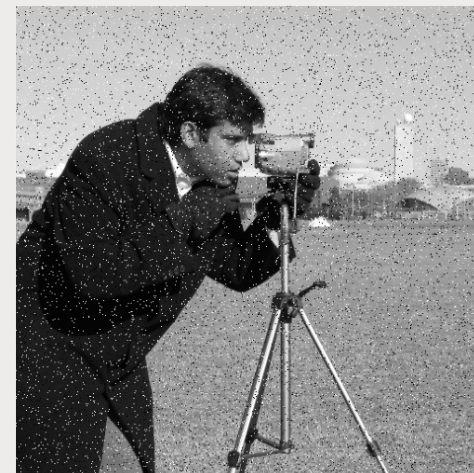
Gaussian noise



Poisson noise

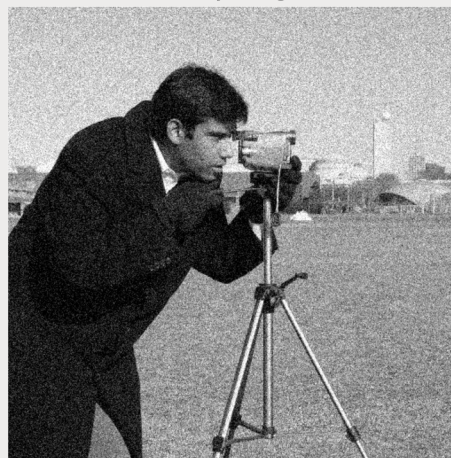


Salt and pepper noise



(c) Cauchy noise

Noisy image



Filtered image





Original Image



Blurring



Sharpening

Deep Learning Approaches

Generative Adversarial Networks (GANs)

- GANs are a type of deep learning framework that involves two competing neural networks – a generator and a discriminator.

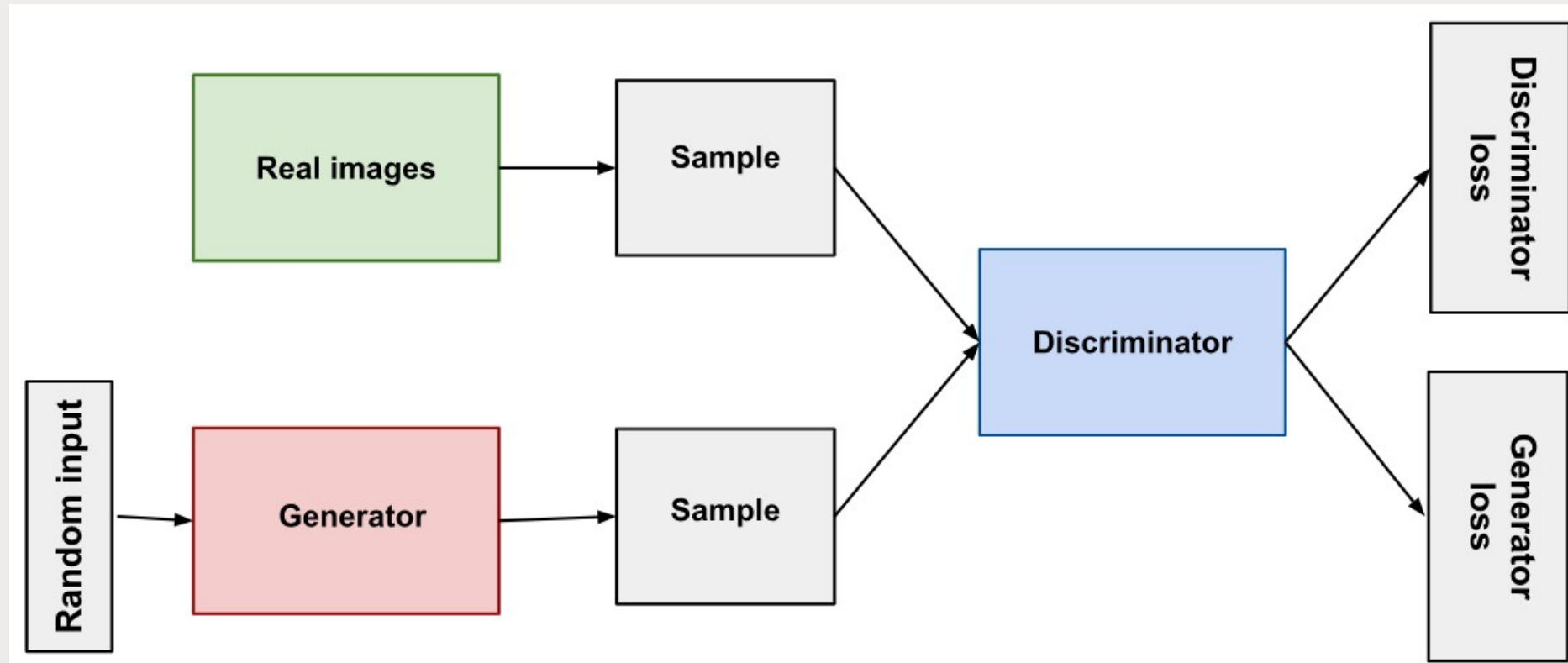
(GAN is a machine learning model that uses two neural networks to create new data)

1) The Generator: This neural network takes in some random noise (like random numbers) as input and tries to convert that into new synthetic data samples that mimic the real data distribution. For example, if you're augmenting images of faces, the generator would try to create brand new fake face images from the noise input.

2) The Discriminator: This neural network takes in both real data samples from the original dataset and the fake samples created by the generator. Its job is to examine each sample and predict whether it is real (from the true dataset) or fake (created by the generator).

Deep Learning Approaches

Generative Adversarial Networks (GANs)



Deep Learning Approaches

Generative Adversarial Networks (GANs)

3) The Training Process: During training, the generator and discriminator play a game, constantly adjusting to outwit each other. The generator aims to create samples that can fool the discriminator into thinking they are real. The discriminator adjusts to get better at spotting fakes from the generator.

4) Over many training rounds, the generator gradually learns to create higher-quality fake samples that are indistinguishable from real data. These synthetic samples can then be used to augment the original dataset.

- GANs can produce highly realistic augmented data across various domains like images, text, audio etc. However, they are complex to train and can sometimes produce unwanted artifacts or failures.

Deep Learning Approaches

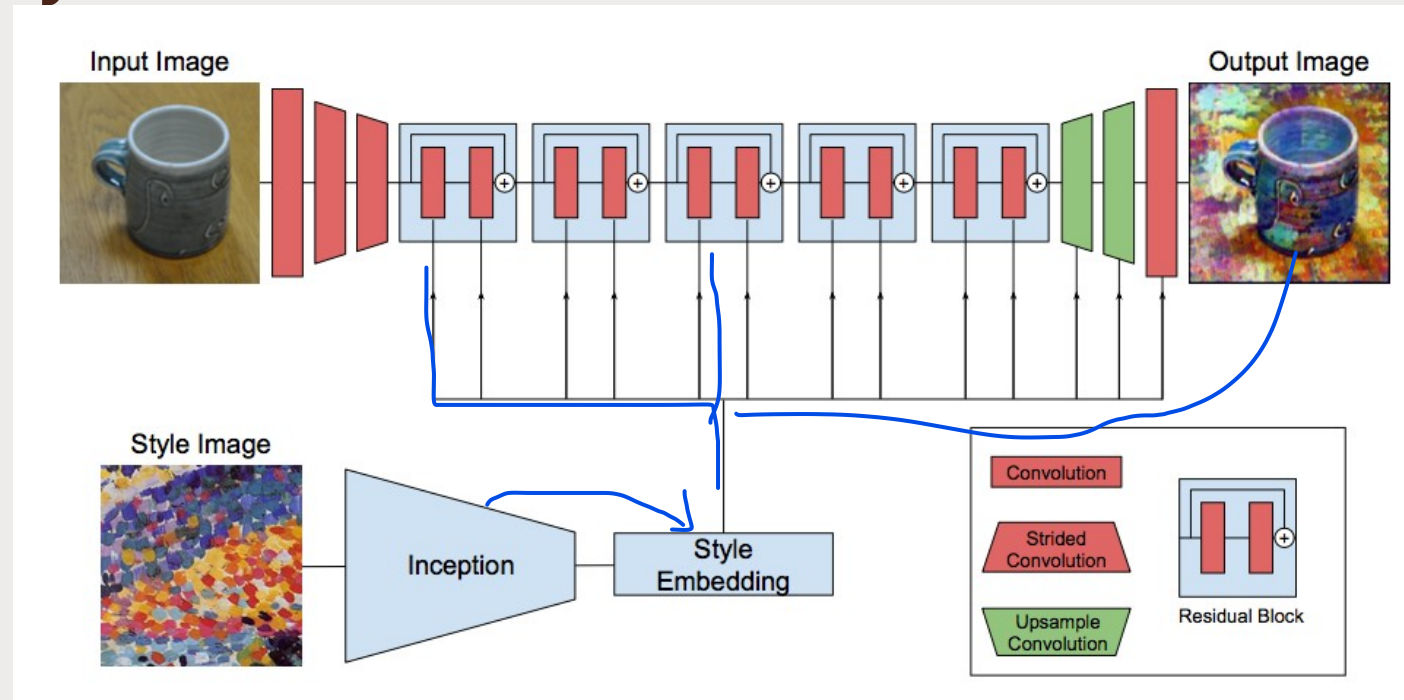
Neural Style Transfer

- Neural Style Transfer is a very exciting deep learning application. The representation created by a series of convolutional layers in a convolutional neural network model can be deconstructed such that content and style can be separated.
- This separation forms the gram matrix, which can be used as a loss function for a generator network, (mapping from a random vector to a height x width x color channel output image).
- This model can be used to do things such as transfer the artistic style of an ordinary photograph.
- Fast Neural Style Transfer “only a single forward pass through a style transfer network
- It is somewhat similar to a color transformation, but it separates itself with the ability to transfer textures and other miscellaneous distortions that are impossible with classic color filters or affine transformations.

“randomizes texture, contrast, and color, while preserving shape and semantic content-preserving transformations to improve the quality of the dataset”

Deep Learning Approaches

Neural Style Transfer



Deep Learning Approaches

Neural Style Transfer



This approach separates and recombines the content and style elements from data samples, especially images. The key steps are:

- 1) A neural network is first trained on a large dataset of images to understand how to separate and encode the content representations (like objects, shapes etc.) from the style representations (like colors, textures, artistic styles etc.)
- 2) Once trained, you can pass in two input images – **a content image and a style image**.
- 3) **The content encoder extracts the content representations from the content image.**
- 4) **The style encoder extracts the style representations from the style image.**
- 5) **These content and style codes are then passed through a decoder network that blends them together into a new output image.**
- 6) The output image contains the same content elements as the original content image, but now stylized with the artistic style extracted from the style image.
 - This allows you to generate unlimited new images by combining content from one image with arbitrary styles from other images. These stylized images can augment the original training data.
 - While powerful for image data, the content/style separation concept can also apply to other data types like augmenting text content with different writing styles.

Deep Learning Approaches

Data Augmentation Neural Networks

- Instead of directly generating augmented samples, these neural network architectures learn optimal transformation functions to apply to input data for augmentation:
 - 1) The model takes in an input data sample (image, text etc.) along with a random transformation code.
 - 2) The neural network applies the specified transformation in a differentiable way to the input, creating an augmented output.
 - 3) During training, the model learns complex, non-linear transformations that maximize performance on the downstream task when applied to training data.
 - 4) At inference, random transformation codes can be passed in to generate unlimited new augmented training examples on the fly.
- These learned augmentation models can produce highly effective data transformations tailored for specific domains/tasks, without manual heuristics. However, training such models requires careful formulation of augmentation policies.
- While more complex, these deep learning augmentation methods can produce highly realistic synthetic data, enabling data augmentation for challenging domains.

Data Augmentation Examples

Text Data Augmentation

- Common approaches for text data include word insertion, deletion, and swapping using thesaurus-based synonym replacement. Back-translation (translating to an intermediate language and back) can rephrase texts naturally.
- Data augmentation makes text models robust to paraphrasing, typos, and the way humans express the same ideas differently. It helps with data scarcity issues for low-resource languages.

Word or sentence shuffling: randomly changing the position of a word or sentence.

Word replacement: replace words with synonyms.

Syntax-tree manipulation: paraphrase the sentence using the same word.

Random word insertion: inserts words at random.

Random word deletion: deletes words at random.

Data Augmentation Examples

Image Data Augmentation

- This is one of the most common use cases, applicable to computer vision tasks like image classification, object detection, etc. Simple transformations like flips, rotations, and crops are frequently used. More advanced techniques synthesize entirely new images using GANs or neural style transfer.
- Data augmentation helps image models become robust to variations like viewpoint, lighting and occlusion. It enables training on underrepresented classes and edge cases that are hard to collect manually.

Geometric transformations: randomly flip, crop, rotate, stretch, and zoom images. You need to be careful about applying multiple transformations on the same images, as this can reduce model performance.

Color space transformations: randomly change RGB color channels, contrast, and brightness.

Kernel filters: randomly change the sharpness or blurring of the image.

Random erasing: delete some part of the initial image.

Mixing images: blending and mixing multiple images.

Data Augmentation Examples

Video Data Augmentation

- Videos combine challenges of images (viewpoints, occlusions) and audio (background noise, accents). Data augmentation borrows techniques across both domains.
- Temporal augmentations
 - Dropping/repeating
 - Speed changes
 - Video style transfer using GANs

Data Augmentation Examples

Audio Data Augmentation

- For speech recognition, data augmentation techniques include adding background noise, and changing pitch, speed, or vocal tract characteristics. This exposes models to more accents, pronunciations, and acoustic environments during training.
- Similar transformations can distort music and other audio signals to augment data for audio classification tasks.

Noise injection: add gaussian or random noise to the audio dataset to improve the model performance.

Shifting: shift audio left (fast forward) or right with random seconds.

Changing the speed: stretches times series by a fixed rate.

Changing the pitch: randomly change the pitch of the audio.

Data Augmentation Examples

Sensor & IoT Data Augmentation

For time series data from sensors, common augmentations

- **Injecting noise**
 - **Resampling**
 - **Time warping**
 - **Adding synthetic anomalies that mimic real-world glitches**
-
- This allows training machine learning models for predictive maintenance, anomaly detection, and similar industrial monitoring applications.
 - The choice of augmentation technique depends on the data modality and end application. Often multiple strategies are combined to amplify their benefits.

Use Cases for Data Augmentation in the Medical Imaging Field

Brain Tumor Segmentation

In this application, the goal is to identify and segment out regions in brain scan images (like MRI scans) that contain tumors. However, acquiring a large, high-quality dataset of labeled brain tumor images is very challenging and expensive, as it requires expert radiologists to manually annotate each scan.

Data augmentation techniques can help artificially increase the diversity and size of the training dataset from the limited available data. Common augmentations used include:

- Geometric transformations like flipping, rotating, and scaling the brain scan images
- Adding noise or variations in intensity values
- Cutting out sections and pasting them elsewhere to simulate tumors

With augmented data, the machine learning model can learn richer representations of what brain tumors look like in different orientations, noise conditions etc. This improves the accuracy of tumor detection and segmentation.

Use Cases for Data Augmentation in the Medical Imaging Field

Differential Data Augmentation for Medical Imaging

- This refers to applying different augmentation strategies for the regions/pixels of interest (like tumors, lesions, etc.) compared to the normal background regions in the medical imagery.
- For example, for a tumor region, aggressive augmentations like heavy noise, flips, scaling, etc. may be applied to ensure the model becomes robust to all tumor appearance variations. But for normal backgrounds, lighter augmentations preserve realistic anatomical representations.
- This nuanced, differential augmentation approach prevents lossy artifacts and distortions while amplifying model learning on critical areas of interest.

Automated Data Augmentation for Labeling Medical Images

- Manually labeling every medical image with dense pixel-wise annotations is extremely labor-intensive. Data augmentation can help generate large pools of realistic synthetic images along with their corresponding ground-truth label maps.
- For instance, GANs can learn to synthesize high-quality radiographic images exhibiting diverse pathologies. These generated images can then be automatically labeled using heuristics based on the generated pathology characteristics.
- This jointly augmented image-label dataset can provide abundant training data for developing highly data-hungry medical imaging AI models like segmentation networks.

Use Cases for Data Augmentation in the Medical Imaging Field

Semi-Supervised Data Augmentation for Medical Imaging

- Often, only a small subset of medical imaging data has accurate manual annotations, while a larger portion remains unlabeled. Semi-supervised learning combines both for data-efficient training.
- Here, different augmentation policies can be applied to the labeled vs unlabeled data subsets to maximize model performance. For example:

Labeled Data: Aggressive augmentations like geometric transforms, noise, color distortions etc. to harden the model to variations

Unlabeled Data: Light augmentations like basic flips and translations to introduce some diversity without distorting underlying anatomy

- By strategically augmenting the different data subsets, deep learning models can effectively leverage all available data sources while being robust to real-world distributions.
- In summary, data augmentation plays a crucial role in the medical imaging domain by synthetically amplifying limited labeled datasets, introducing realistic variations that improve model robustness, and enabling data-efficient techniques to maximally utilize all available data resources.