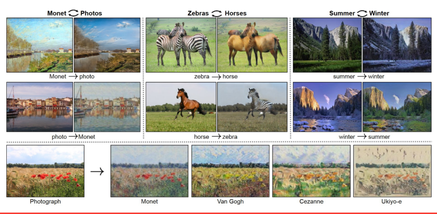
Day 15: CycleGAN

Introduction



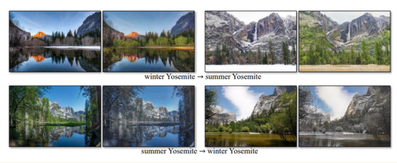
- Learn the mapping between an input image and an output image using a training set of aligned image pairs
- For many tasks, paired training data will not be available, that is where CycleGANs help us
  - Unsupervised Img2img translation

Applications

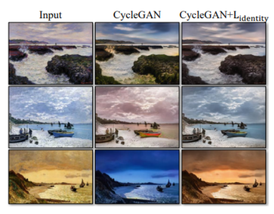
- Collection Style Transfer

- Object Transformation

- Season Transfer



- Photo Generation from Painting



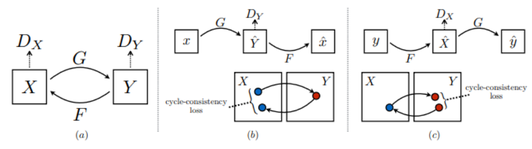
- Photo enhancement

Working

- First take an image input (x) and using the generator G
  - Convert into the reconstructed image
- Reverse this process from reconstructed image to original image using a generator F.
- Calculate the mean squared error loss between real and reconstructed image
- Most important feature

  - Cycle_GAN can do this image translation on an unpaired image where there is no relation exists between the input image and output image

Architecture



- There are 2 generators (G and F) and 2 discriminators (D_X and D_Y) being trained here.
- Generator G learns to transform image X to image Y. (G:X->Y)
- Generator F learns to transform image Y to image X. (F:Y->X)

$$G : X \rightarrow Y$$
$$F : Y \rightarrow X$$

- Discriminator D_X learns to differentiate between image X and generated image X (F(Y)).
- Discriminator D_Y learns to differentiate between image Y and generated image Y (G(X)).

Steps involved

- Set up the input pipeline
  - Install **tensorflow_examples** - importing the generator and the discriminator
- Input Pipeline
  - Image augmentation techniques to avoid overfitting
    - Random Jittering
      - resized to 286 x 286
      - cropped to 256 x 256
    - Random Mirroring
      - randomly flipped horizontally i.e., left to right.
- Import and reuse the Pix2Pix models
  - Import the generator and the discriminator used in Pix2Pix via the installed tensorflow_examples package.
- Loss functions
  - no paired data to train on
    - there is no guarantee that the input x and the target y pair are meaningful
  - To enforce that the network learns the correct mapping, the authors propose the cycle consistency loss.
  - In Cycle consistency loss
    - Image X is passed via generator G that yields generated image Y^.
    - Generated image Y^ is passed via generator F that yields cycled image X^.
    - Mean absolute error is calculated between X and X^
    - forward cycle consistency loss:X->G(X) ->F(G(X))~X^

    - backward cycle consistency loss:Y->F(Y) ->G(F(Y))~Y^

  - Identity Loss
    - If you fed image Y to generator G, it should yield the real image Y or something close to image Y.
    - Identity loss=|G(Y)−Y|+|F(X)−X|
  - Initialize the optimizers
- Checkpoints
- Training
  - Four basic steps:
    - Get the predictions.
    - Calculate the loss.
    - Calculate the gradients using backpropagation.
    - Apply the gradients to the optimizer.

Hands-on