Slide-8-Style-transfer

Created	@Apr 18, 2021 2:32 PM

TFRecord

▼ What is TFRecord?

- The recommended format for TF
- Binary format

▼ Why use binary in TFRecord?

- make better use of disk cache
- faster to move around
- can handle data of different types (images and lables in one place)

▼ Flow of TFRecord?

Step 1: You convert data to TFRecord format

Step 2: Using `TFRecordDataset` to read TFRecord

Style Transfer

▼ The core idea?

- Whose content is closest to the content image

- Whose style is closet to the style image

▼ What layers of ConvNet related to content/style of and image?

- lower layer extract feature related to content
- higher layer extract feature related to style

▼ The loss function using here?

- `Content loss`: Measure the content loss between the content of the generated image and the content of the content image
- `Style loss`: Measure the style loss between the style of the generated image and the style of the style image

⇒

- `Content loss`: Measure the loss between the feature maps in the content layer
- `Style loss`: Measure the loss between the feature maps in the style layer

▼ How to find these magic feature maps?

- Using pretrained weights (function) such as VGG, AlexNet, GoogleNet

▼ Detail of Content loss?

$$\mathcal{L}_{content}(\vec{p}, \vec{x}, l) = \frac{1}{2} \sum_{i,j} \left(F_{ij}^{l} - P_{ij}^{l}\right)^2$$

▼ Detail of Style loss?

$$E_l = \frac{1}{4N_l^2 M_l^2} \sum_{i,j} \left(G_{ij}^l - A_{ij}^l\right)^2$$

$$\mathcal{L}_{style}(\vec{a}, \vec{x}) = \sum_{l=0}^{L} w_l E_l$$

▼ Detail of total loss?

$$\mathcal{L}_{total}(\vec{p}, \vec{a}, \vec{x}) = \alpha \mathcal{L}_{content}(\vec{p}, \vec{x}) + \beta \mathcal{L}_{style}(\vec{a}, \vec{x})$$

▼ What is the trick when implement?

- Train input instead of weights
- Multiple tensors share the same variable to avoid assembling identical subgraphs
- Use pre-trained weights (from VGG-19)