Image Corlorization via Deep-learning

Khoa Tran Nhat

Linh Ly Nguyen Thuy

University of Information Technology

9th December 2024

[Link github]

Motivation

- Cultural and Historical Preservation
- Teaching and Visualizing Information.
- Data Augmentation Based on Color.

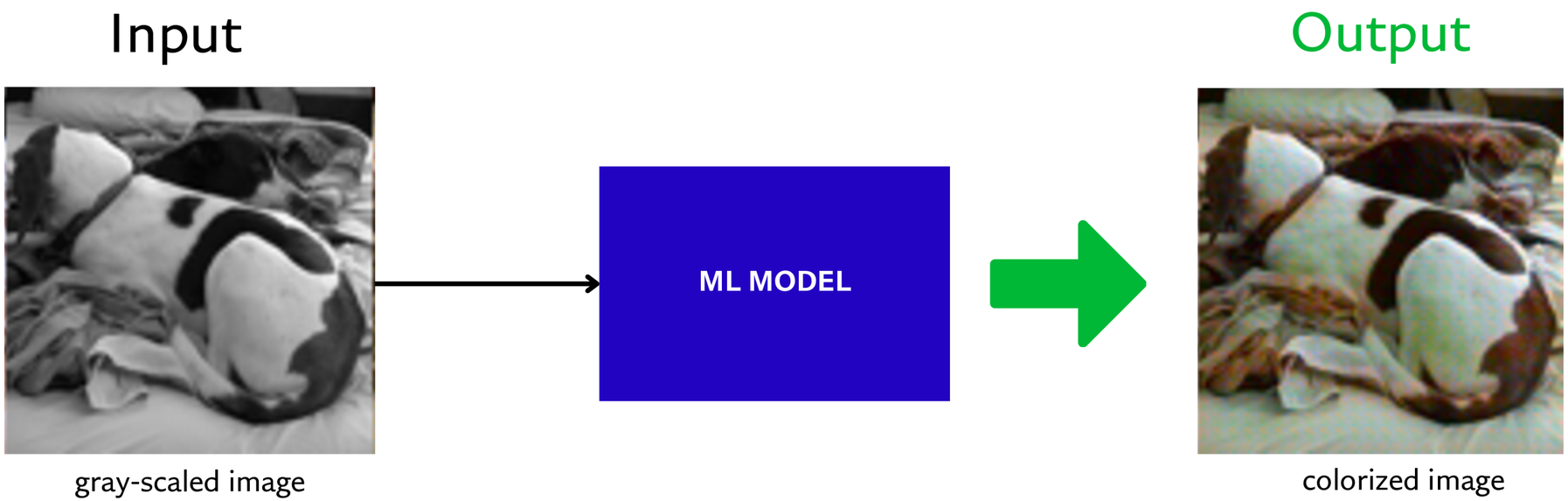
Before and after applying image colorization on historical images.

Before and after applying image colorization on children drawing.

Data argument.

Problem Statement

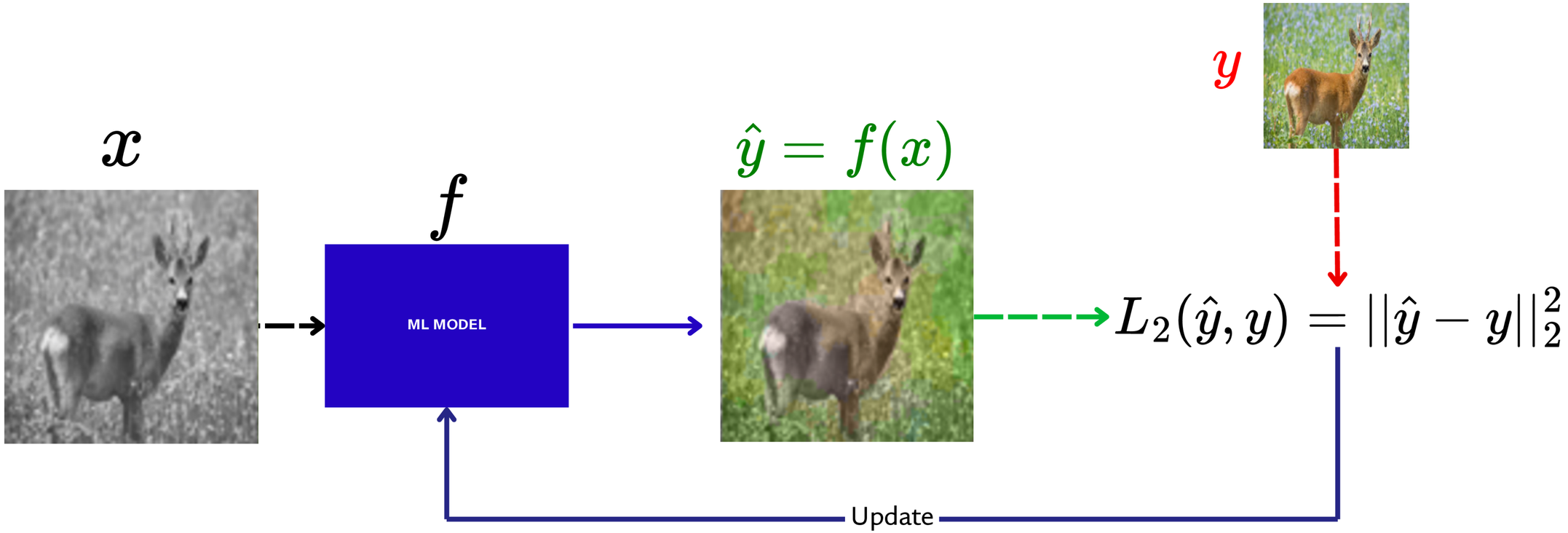
- Inference level.
  - **Input:** a **gray-scaled image** (sketch image without colors).
  - **Output:** a **colorized version** of an input image.
- Training level.
  - **Input: dataset gray-scaled images** and **original colorized images**.
  - **Output: The trained model**.



Problem statement - Inference level.

Methodology: Reconstruction Approach

- A reconstruction model that takes a grayscale **single-channel** image as input and reconstructs it into an **RGB** image $f : \mathbb{R}^{H \times W \times 1} \rightarrow \mathbb{R}^{H \times W \times 3}$.



Recontruction baseline with RGB approach.

Methodology: Reconstruction Approach

- A reconstruction model that takes a grayscale **single-channel** image as input and reconstructs it into an **RGB** image $f : \mathbb{R}^{H \times W \times 1} \rightarrow \mathbb{R}^{H \times W \times 3}$.
- But this approach has limitations:
  - The output image **must reconstruct** the entire image, including *edges*, *structure, and other details*.
  - The method involves regression from 1 value to 3 values, whereas we only have **one grayscale channel** and **need to predict two additional values**.

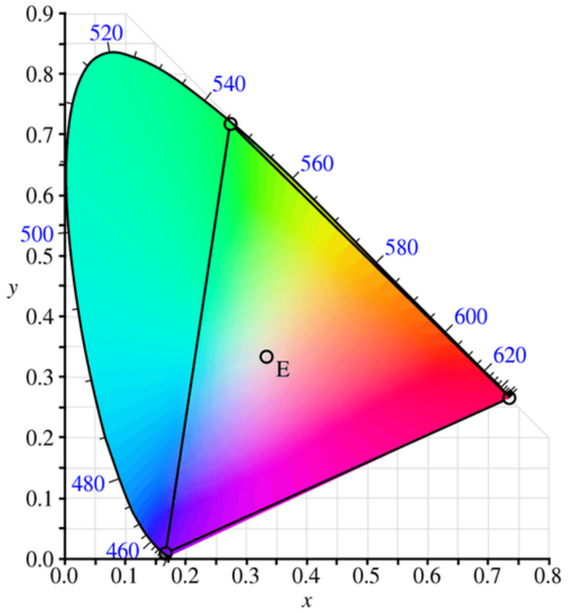
x $\hat{y} = f(x)$

Fig.5: 200 epochs Visualization with with Recontruction approach - RGB channel.

Methodology: Reconstruction Approach

- A reconstruction model that takes a grayscale **single-channel** image as input and reconstructs it into an **RGB** image $f : \mathbb{R}^{H \times W \times 1} \rightarrow \mathbb{R}^{H \times W \times 3}$.
- But this approach has limitations:
  - The output image **must reconstruct** the entire image, including *edges*, *structure, and other details*.
  - The method involves regression from 1 value to 3 values, whereas we only have **one grayscale channel** and **need to predict two additional values**.

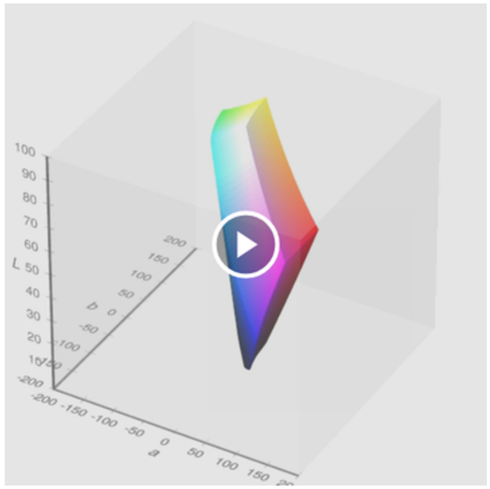
CIE Lab Color Channel



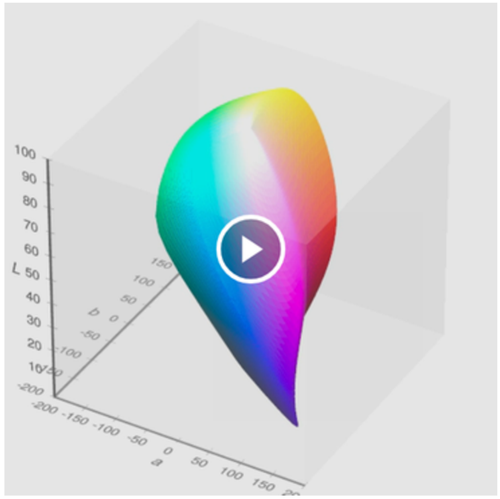
Methodology: CIE LAB Color Model

- **The CIELAB** is a color model that was designed to be **device-independent** and closely approximate **human vision**. It stands for:
  - L (lightness) $\in (0, 100)$
  - A (green to red) $\in (-128, 128)$
  - B (blue to yellow) $\in (-128, 128)$

- A reconstruction model that takes a single-channel grayscale image **L** as input and predicts two output values: the **a** and **b** components. $f : \mathbb{R}^{H \times W \times 1} \rightarrow \mathbb{R}^{H \times W \times 2}$

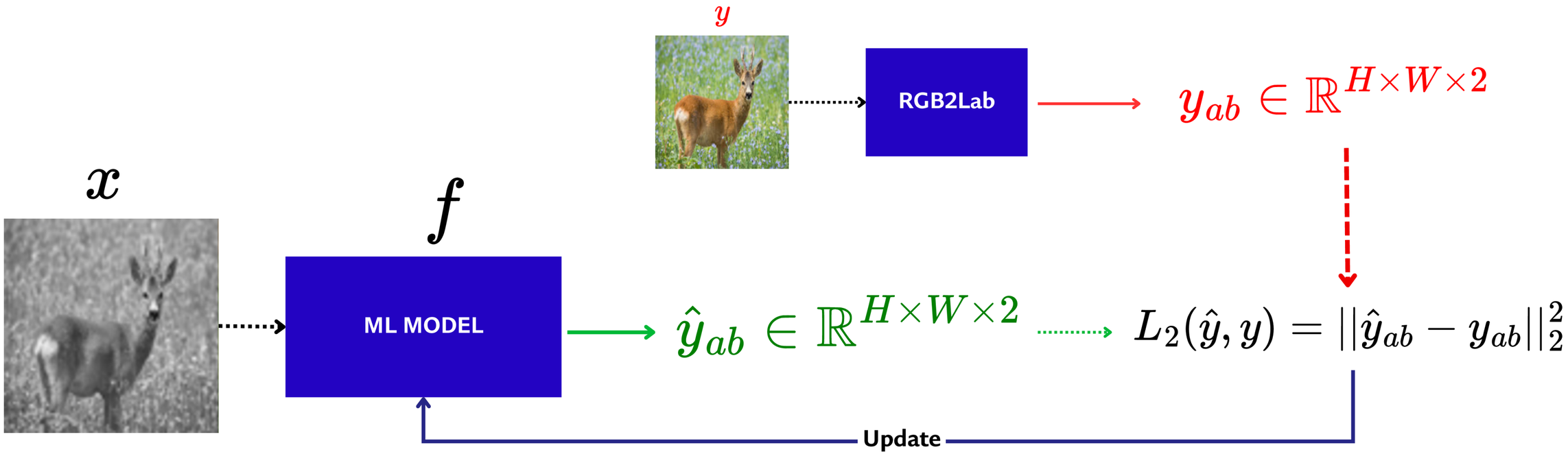


SRGB gamut within CIELAB color space



Visible gamut within CIELAB color space

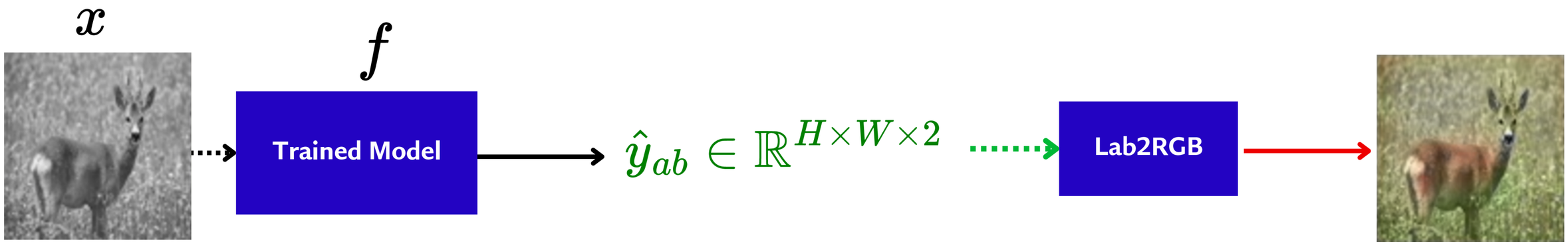
Methodology: CIE LAB Color Model



Training - CIE Reconstruction Training approach.

Methodology: CIE LAB Color Model

- To perform inference, the **grayscale image** is passed through the model to generate the a and b channels. **These channels are then merged with the grayscale image** to reconstruct the colorized image.



Inference - CIE Reconstruction approach.

Methodology: Classification Approach

- With the reconstruction approach, we just return the **single option (a,b)**.
- We aim to determine **the likelihood** that a specific pixel is colorized with a particular color.

Predicting the **probability distribution** over possible color values for each pixel.

- A classification model that takes a grayscale **single-channel** image as input and return the probability of all posible colors.

Methodology: Classification Approach

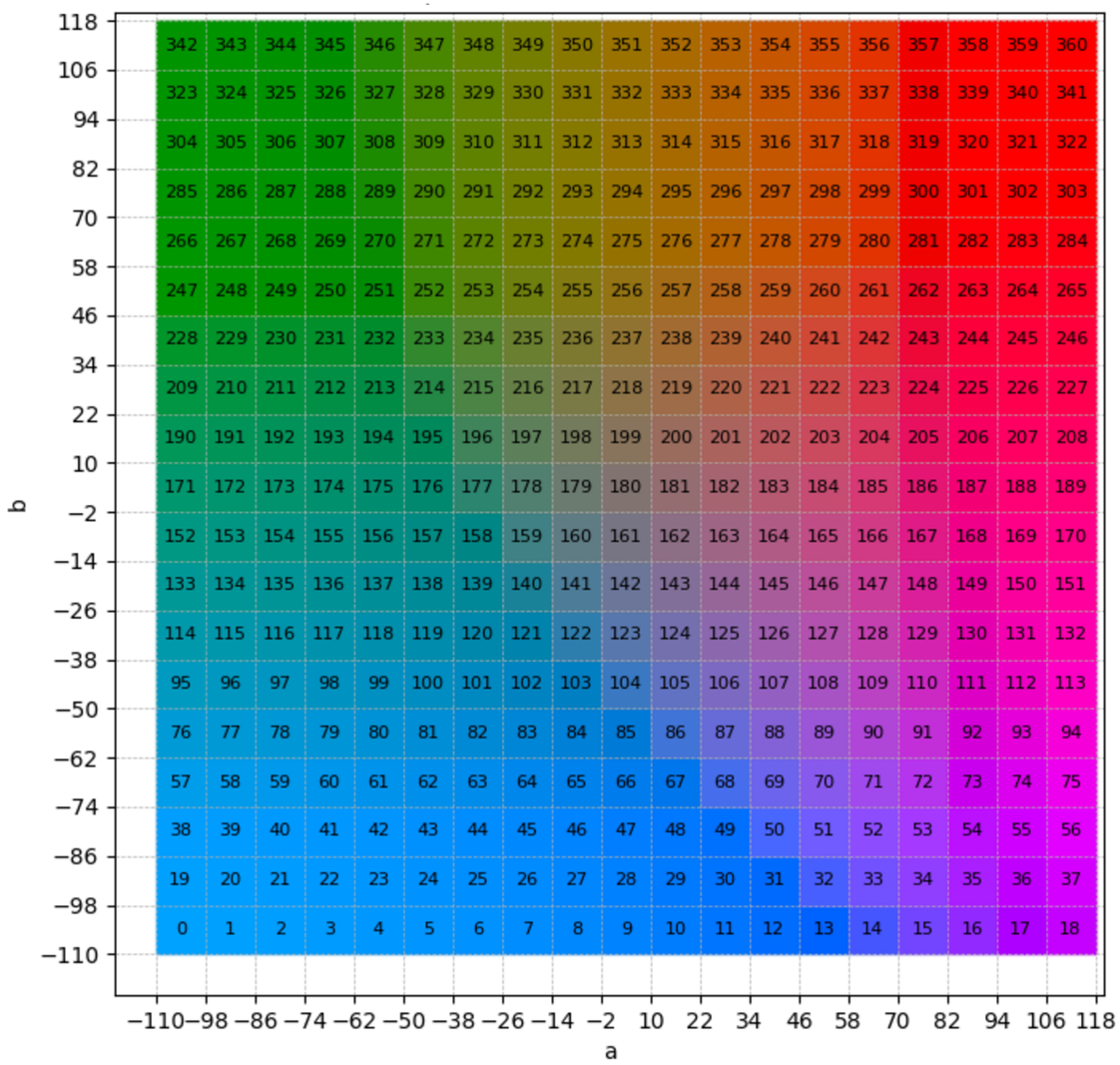
- But this approach must convert the continueous space to the categaory space.
- A simple approach is to divide the RGB color space into **Q** classes, where each class represents a **fixed-size color bin**.

$$f : \mathbb{R}^{H \times W \times 1} \rightarrow \mathbb{R}^{H \times W \times Q}$$

- In RGB space: $Q = \dfrac{256^3}{binsize^3}$
- Using **the CIE Lab color space**, we focus only on the **a** and **b** channels to simplify the color space.

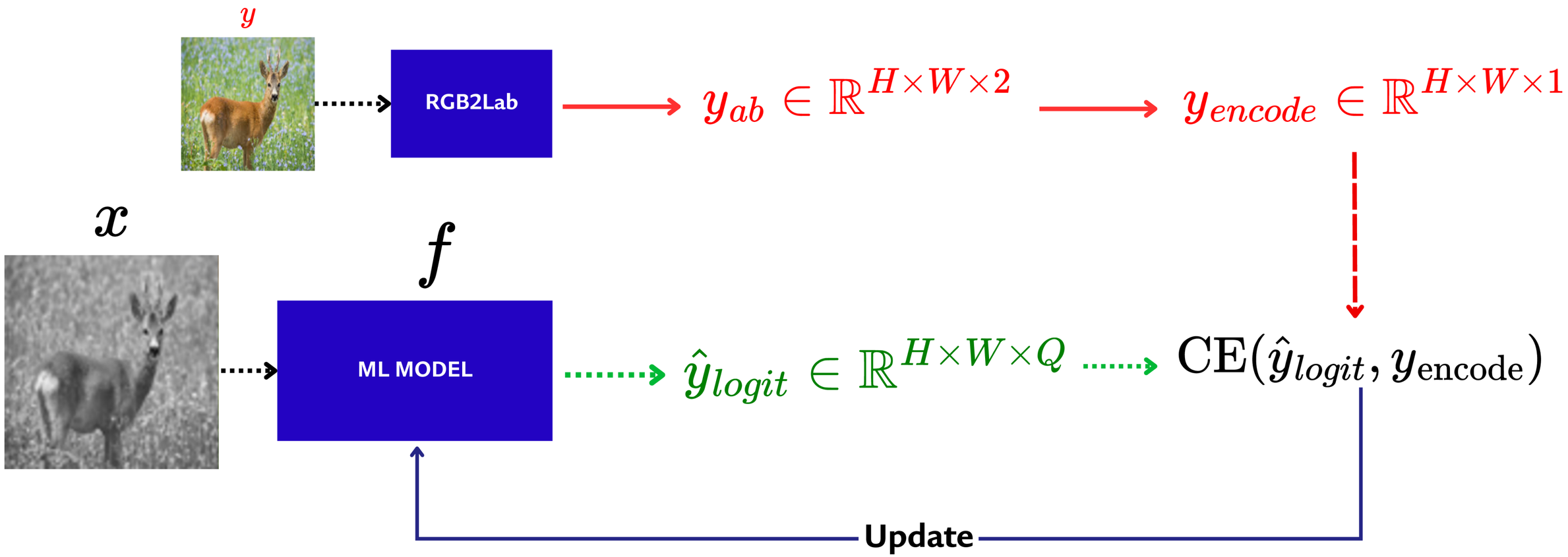
Methodology: Classification Approach

- With **bin size = 12**, and limit the a,b beetween **[-110, 118]**, we archive **Q = 361**.



Visualization of the **Q classes** in the **ab space**, with the **RGB centroid** assigned to each bin for each class.

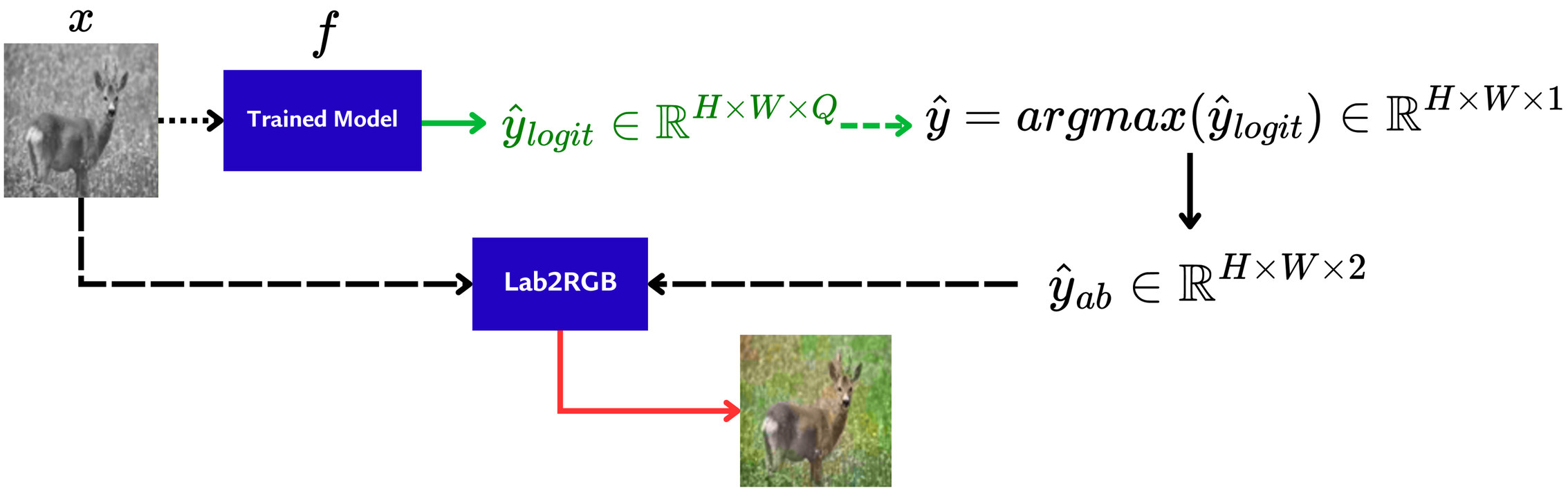
Methodology: Classification Approach



Training - CIE Classification Training approach.

Methodology: Classification Approach

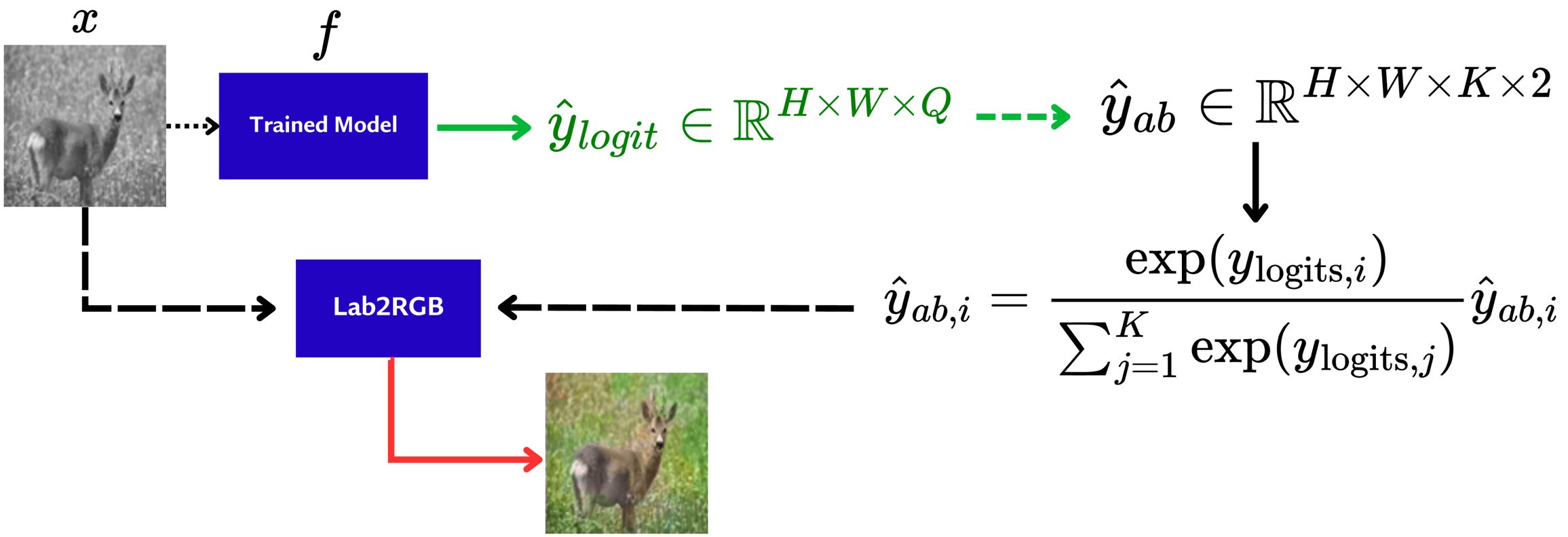
- The initial idea is to select the bins with the **highest likelihood** and **assign their centroid (ab)** values to the corresponding pixels.



Inference(centroid approach) - CIE Classification Training approach.

Methodology: Classification Approach

- Using the top-k predicted classes, we convert them into ab values.
  The final ab values are calculated as **the weighted sum of the top-k** ab values, with weights given by their **softmax probabilities**.



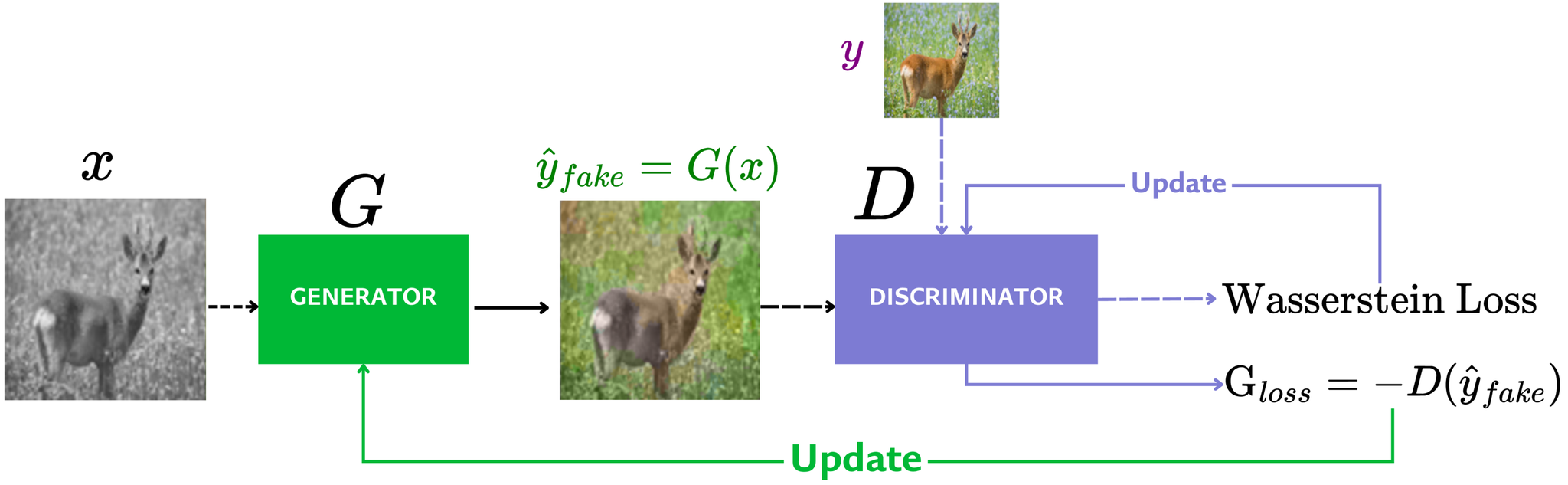
Inference (softmax approach) - CIE Classification Training approach.

Methodology: GAN

- The main target of colorization is to color **reasonably** and maybe **creatively**.

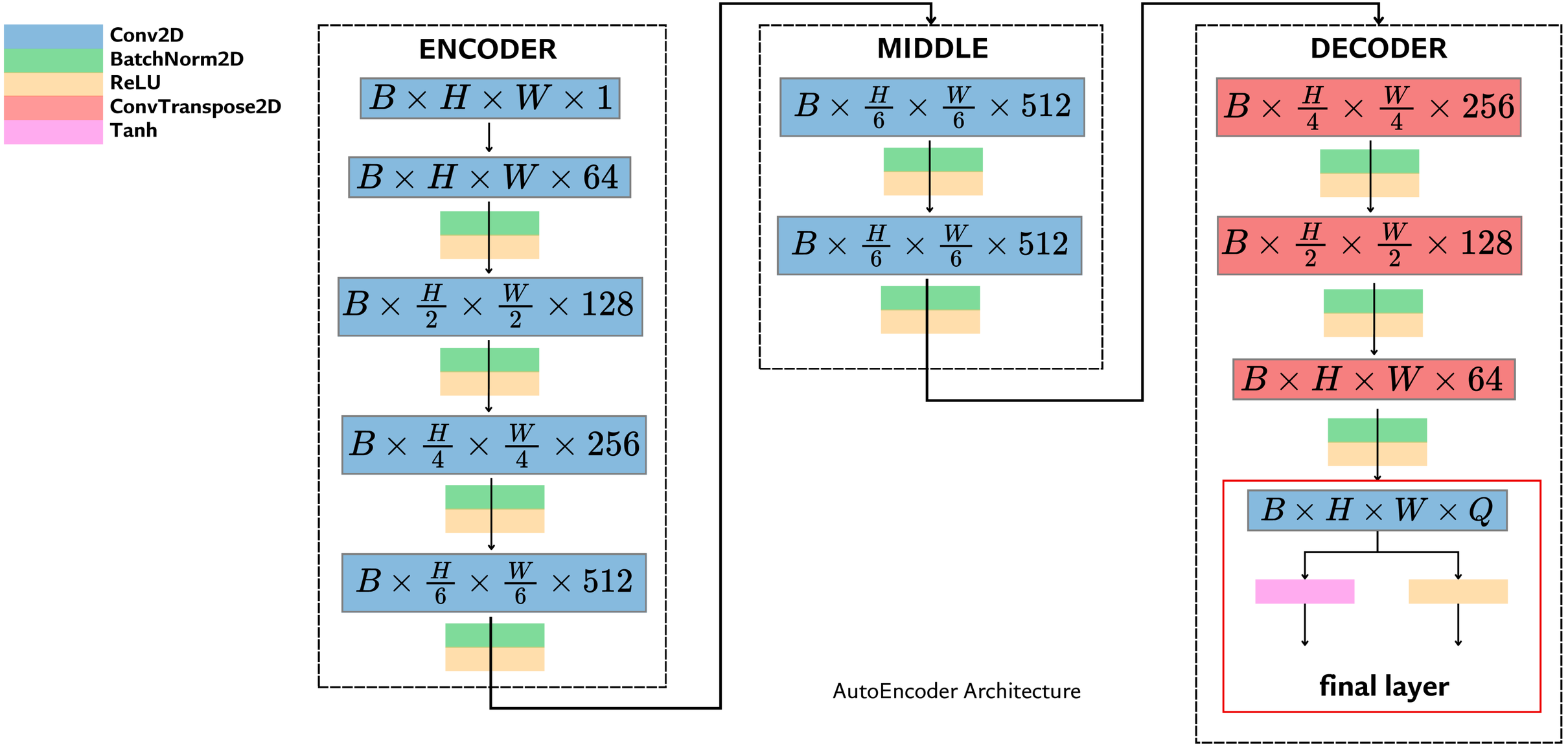
  Colorization can be approached using **GANs**, where the **generator creates colorized images**, and to deceive the discriminator, the generator will try to color **as naturally as possible.**

- To be “creative”, or preventing mode collapse and vanishing gradient, we chose **WGAN-GP** as GAN approach.



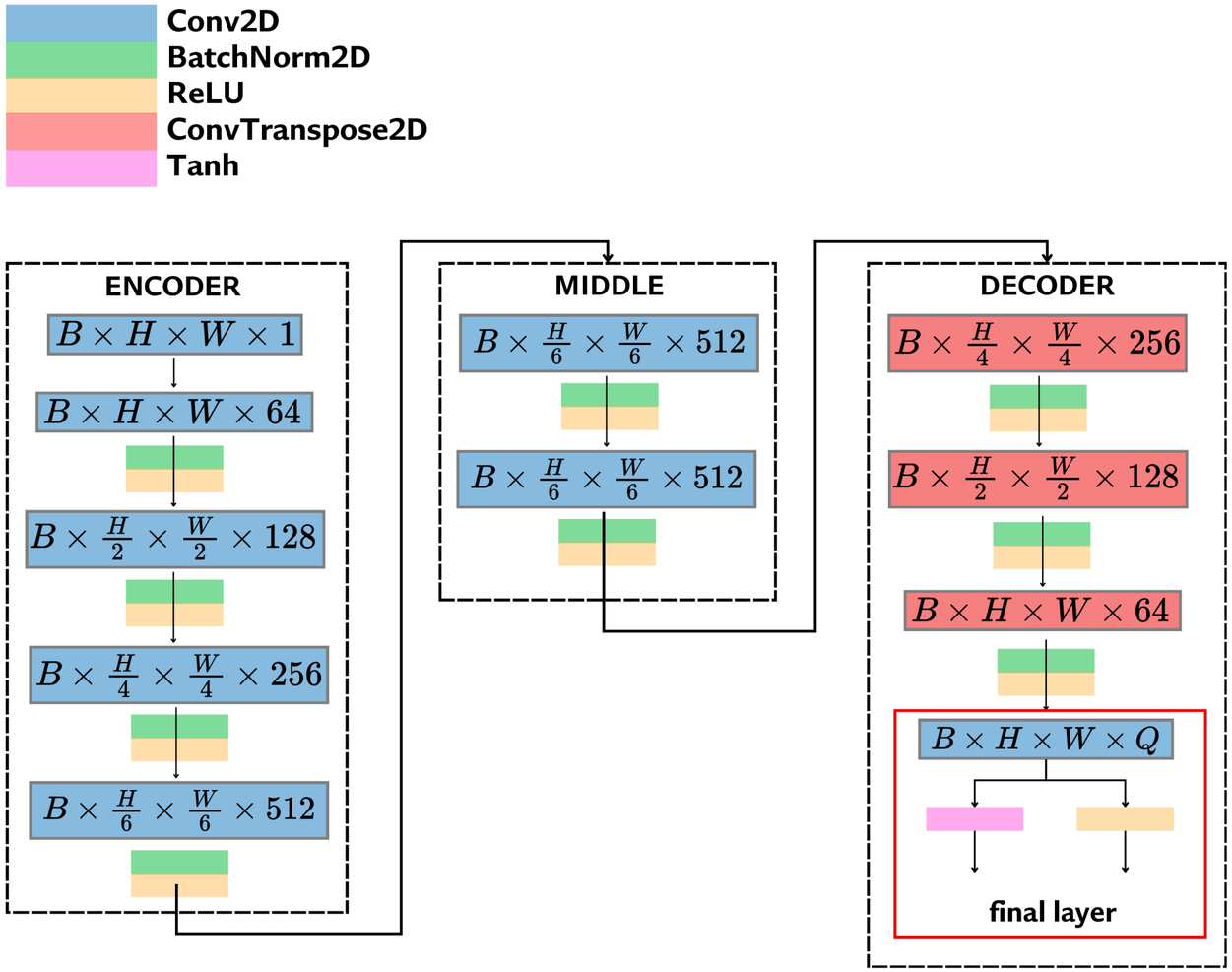
WGAN - CIE Reconstruction Training approach.

Architecture: AutoEncoder

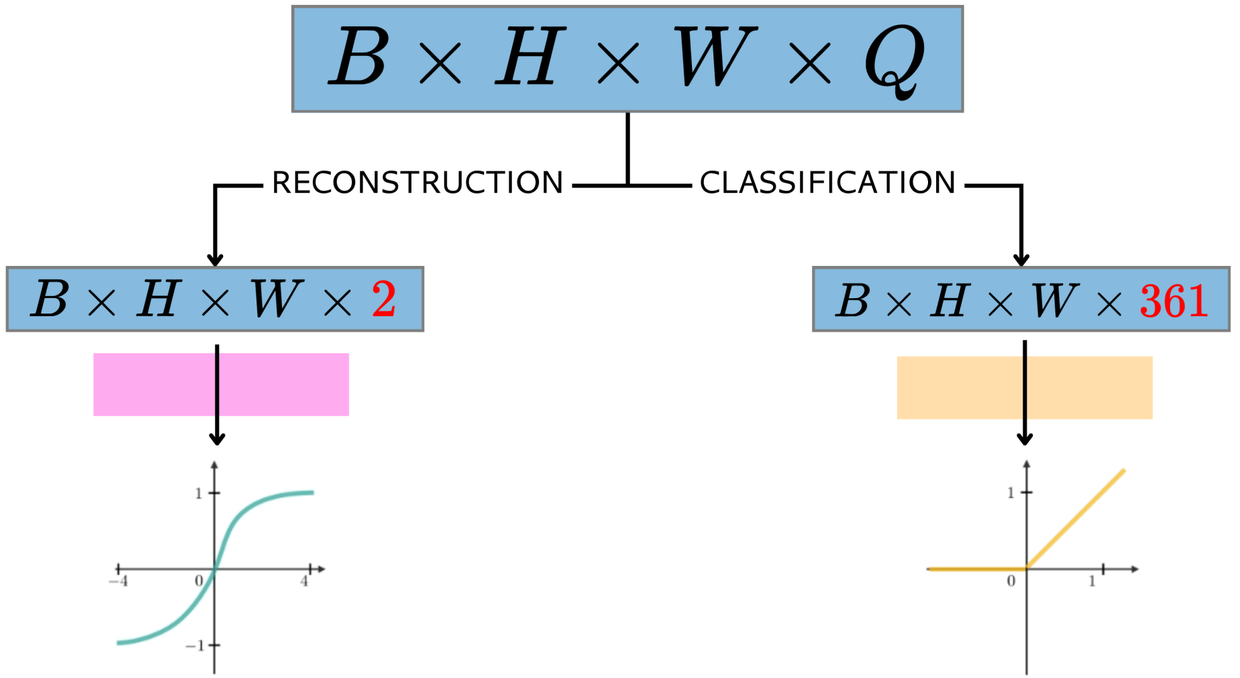


AutoEncoder Architecture

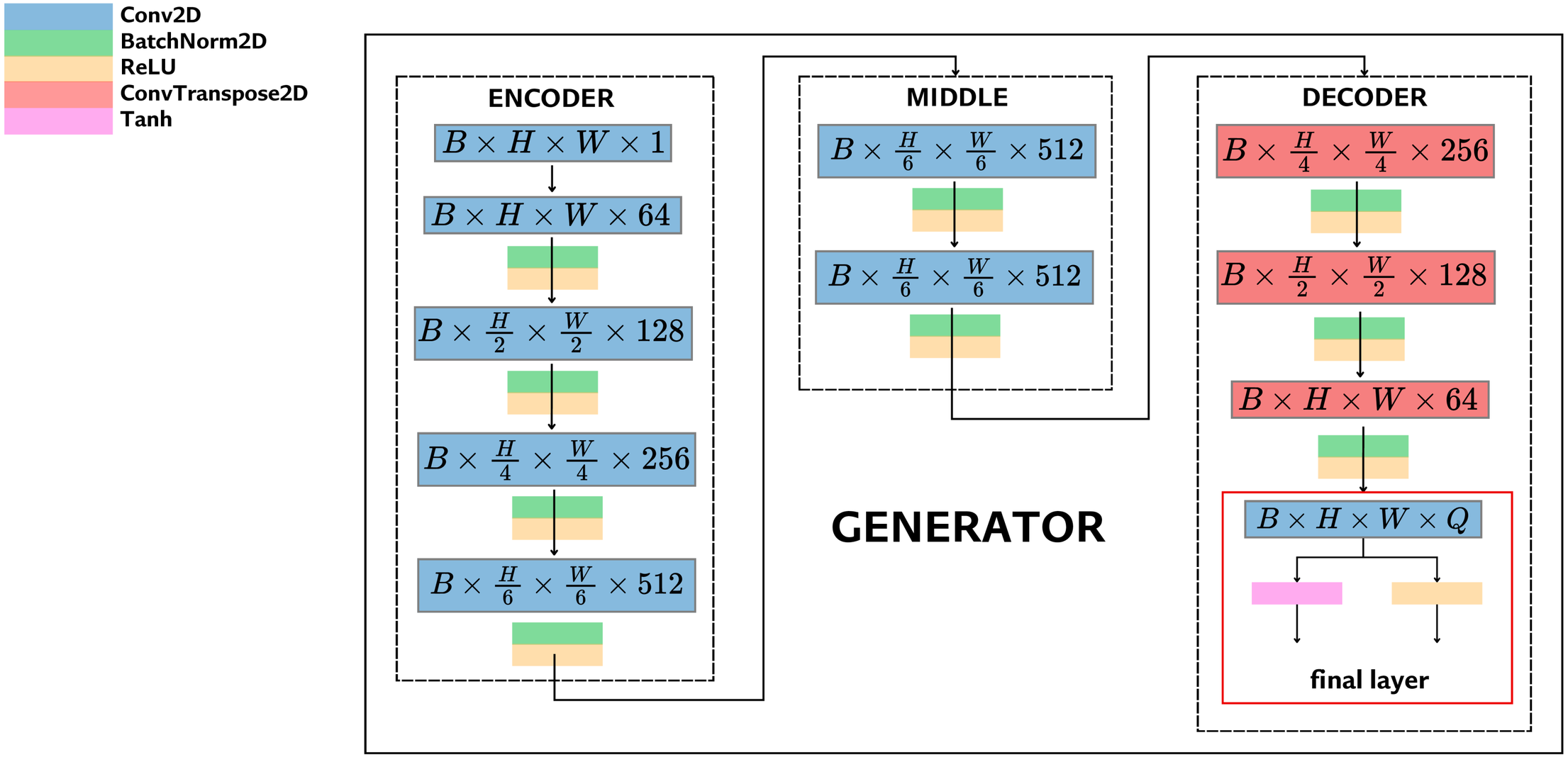
Architecture: AutoEncoder



AutoEncoder Architecture



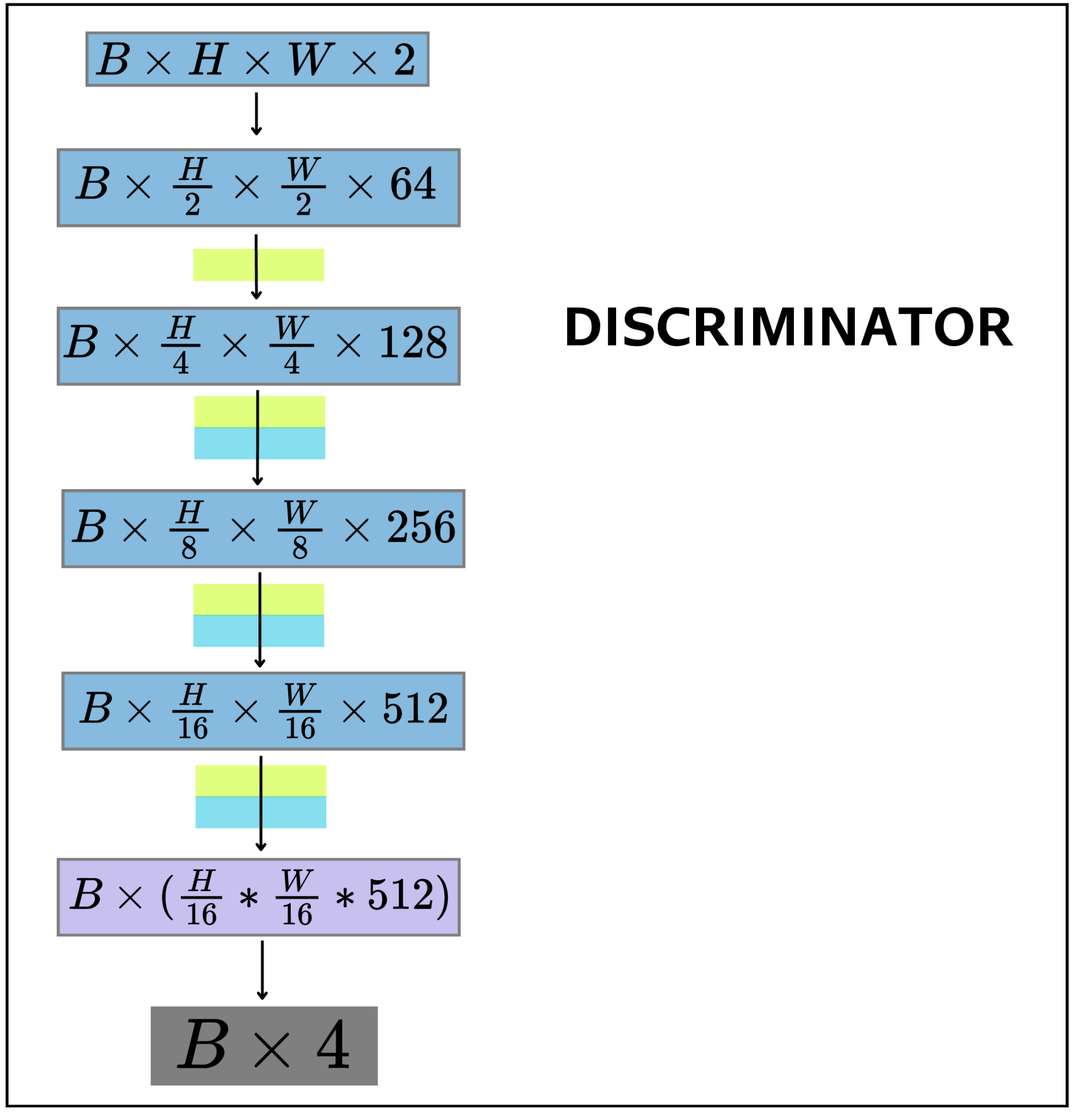
Architecture: GAN



GAN Generator

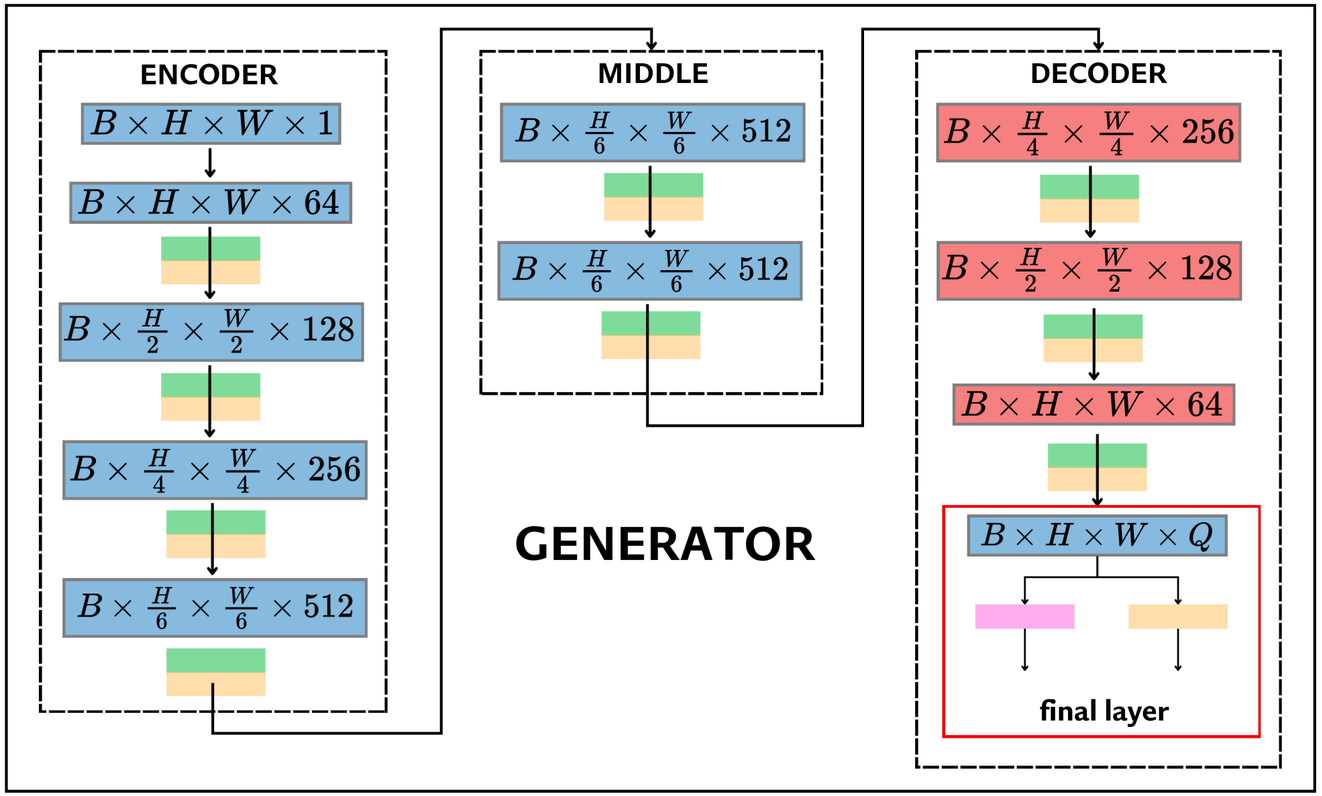
Architecture: GAN

Conv2D
LeakyReLU (0.2)
InstanceNorm2d
Flatten
Linear

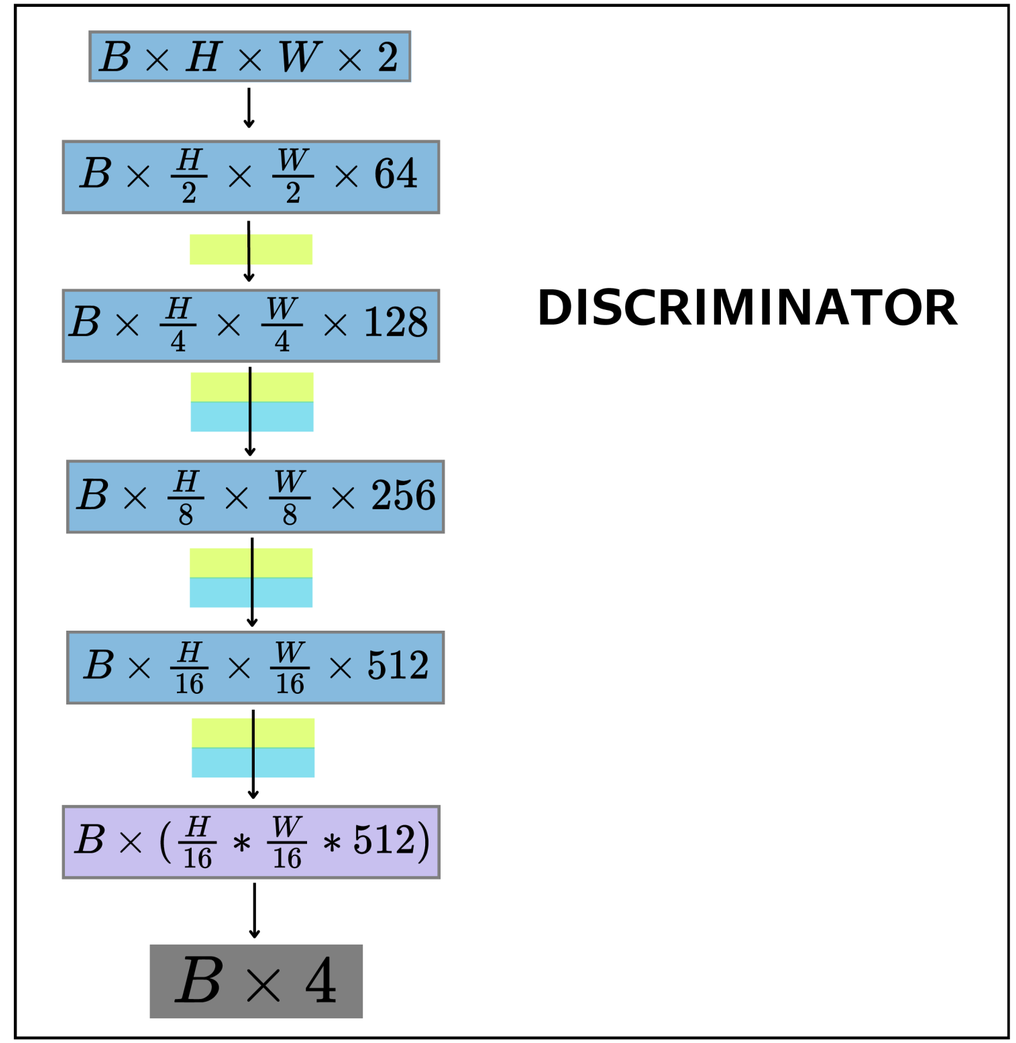


GAN Discriminator

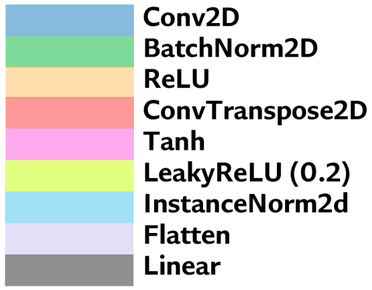
Architecture: GAN



GAN Generator



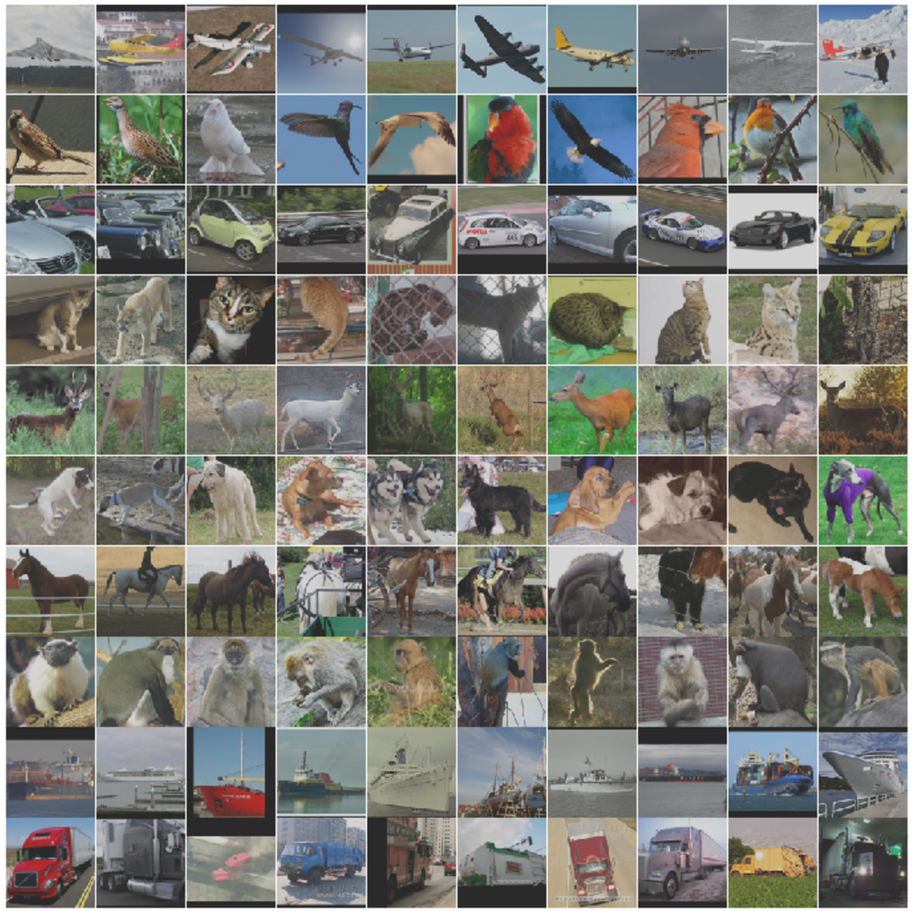
GAN Discriminator



Experiment

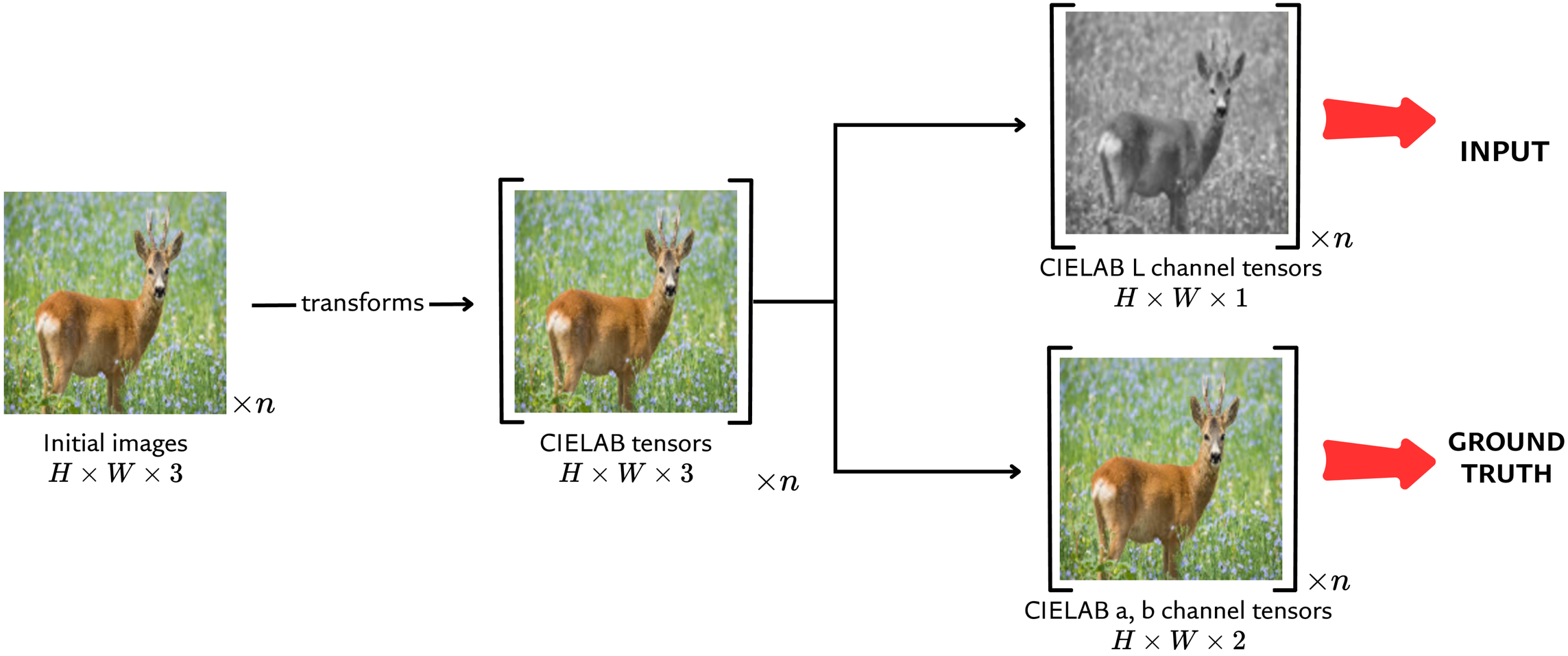
Dataset & Data Processing

- **STL10 - Dataset**: object image dataset for classification task including **96x96** images, divided into 10 classes.
  - Training set (unlabeled dataset): 100,000 images
  - Testing set: 8,000 images



Dataset & Data Processing

- **CIELAB** dataset:



Evaluation: Peak Signal-to-Noise Ratio (PSNR)

Mean Square Error (MSE)

$$MSE = \frac{1}{MxN} \sum_{x=1}^{M} \sum_{y=1}^{N} [f(x,y) - g(x,y)]^2$$

where:
MxN: total number of pixels in the image
$f(x,y)$: input image
$g(x,y)$: enhanced (output) image

Peak Signal-to-Noise Ratio (PSNR)

$$PSNR = 10 \log \frac{f_{max}^2}{MSE}$$

where:
$f_{max} = 255$ is the maximum gray value

Evaluation: Inception Score (IS)

Inception Score (IS)

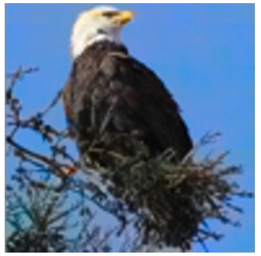
$$\text{IS}(G) = \exp\left(\mathbb{E}_{x \sim p_{\text{data}}}\left[D_{\text{KL}}\left(p(y|x) \parallel p(y)\right)\right]\right)$$

- $\text{IS}(G)$: Inception Score of the generative model G.
- p_{data}: The real data distribution.
- $p(y|x)$: The predicted class distribution from the Inception model on sample x.
- $p(y)$: The marginal distribution of the predicted class labels across all samples.
- D_{KL}: The Kullback-Leibler Divergence, which measures the difference between the predicted distribution $p(y|x)$ and the average class distribution $p(y)$.
- exp: The exponential function, representing the average of the logarithm.

Experiment Result

		PSNR	IS
RECONSTRUCTION		**30.44**	12.1235
CLASSIFICATION	centroid	25.2813	18.6873
	softmax - assigned	25.4372	**18.7483**
GAN		23.2057	17.1104

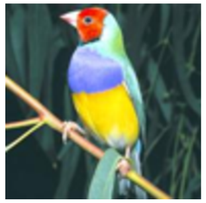
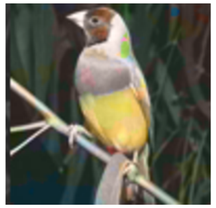
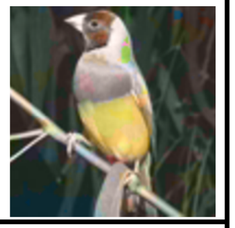
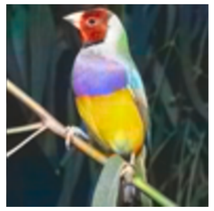
Inference Result

INPUT	RECONSTRUCTION	CLASSIFICATION - CENTROID	CLASSIFICATION - SOFTMAX	GAN	GROUNDTRUTH
					
					
					

Inference Result

INPUT	RECONSTRUCTION	CLASSIFICATION - CENTROID	CLASSIFICATION - SOFTMAX	GAN	GROUNDTRUTH
					
					
					

Limitations

INPUT		**GROUND TRUTH** 
RECONSTRUCTION		• Pros: ◦ stable and easy training convergence ◦ easy implementation • Cons: lack of creativity
CLASSIFICATION - centroid		• Pros: ◦ more creative than reconstruction • Cons: ◦ larger architecture ◦ slower to convergence
CLASSIFICATION - softmax		
GAN		• Pros: ◦ most creative of all • Cons: ◦ hard to convergence due to mode collapse or other insufficient training stability

- Performance on real-world images is limited, as the STL10 dataset lacks generalization and includes noisy data (e.g., grass backgrounds), biasing colorization toward greenish tones.

References

[1] Richard Zhang, Phillip Isola, Alexei A. Efros: Colorful Image Colorization. ECCV (3) 2016: 649-666

[2] Karen Simonyan, Andrew Zisserman, ‘Very Deep Convolutional Networks for Large-Scale Image Recognition’. ICLR 2015

[3] Adam Coates, Honglak Lee, Andrew Y. Ng An Analysis of Single Layer Networks in Unsupervised Feature Learning AISTATS, 2011.

[4] https://github.com/mjhorvath/Datumizer-Wikipedia-Illustrations

[5] Luo, M.R. (2015). CIELAB. In: Luo, R. (eds) Encyclopedia of Color Science and Technology. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-27851-8_11-1

THANKS FOR WATCHING!