Machine Learning Hackathon: Image Compression System for Autonomous Underwater Vehicles (AUVs)

K.Noniesh Reddy
K.Sai Dheeraj
B.Lokeshwar

Introduction:

Image compression is a fundamental task in the field of computer vision and image processing. It involves reducing the size of an image while preserving its essential information.We used CAEs for this challenge. Convolutional Autoencoders (CAEs) are a class of neural networks that have shown promising results in image compression tasks. By leveraging convolutional layers for feature extraction and reconstruction, CAEs can effectively compress images while maintaining their visual fidelity.

Objective:

The objective of this report is to explore the application of Convolutional Autoencoders for image compression. We aim to demonstrate how CAEs can be trained to compress images into a lower-dimensional latent space representation and then reconstruct them with minimal loss of quality.We need to compress it at least 300 times.

Methodology:

1.Data Preparation:We begin by using the dataset that is given to us..we used paired images folder which has 3 subfolders

Dataset Name	Training Pairs	Validation	Total Images
Underwater Dark	5550	570	11670
Underwater ImageNet	3700	1270	8670
Underwater Scenes	2185	130	4500

We trained our model with these datasets.We reshaped the images to **(224,224,3)** shape.After reshaping we normalized the images between [0,1].We divided the dataset into 3 parts: train,test and validation data sets.We divided them according to dataset size.
We used convolutional autoencoders for this task.because we need to Compress it at least 300 times restoring its spatial information so used Autoencoders as we are dealing with images we used CAE for Dimensionality reduction and reconstruction.

2.CAE(Convolutional Auto Encoder) Architecture:

Encoder:

Input: The input shape is (224, 224, 3), indicating an image with a height and width of 224 pixels and 3 color channels (RGB).

Convolutional Layers:

- The first convolutional layer (conv1) has 6 filters of size 5x5 with ReLU activation and padding to maintain the spatial dimensions.
- It's followed by a max-pooling layer (maxpool1) with a pool size of 2x2 to reduce spatial dimensions while preserving important features.
- The second convolutional layer (encoder1) has 12 filters of size 5x5 with ReLU activation and padding.
- Another max-pooling layer (maxpool2) follows with the same configuration as before.
- The third convolutional layer (coding) has 16 filters of size 5x5 with ReLU activation and padding.

Flattening and Dense Layers:

- The output of the last convolutional layer is flattened (flatten) to prepare for the fully connected layers.
- There are three fully connected layers (hidden_1, hidden_2, hidden_3) reducing the dimensionality of the encoded features gradually from 9408 to 500.

Decoder:

Dense Layers:

- Three fully connected layers (hidden_4, hidden_5) increase the dimensionality back to 9408.
- The output is reshaped (reshaped) to match the shape before flattening.

Transposed Convolutional Layers:

- In the context of autoencoders, transposed convolutional layers are commonly used in the decoder part to reconstruct the original input from the compressed representation learned by the encoder.
- Two transposed convolutional layers (decoder2, decoder1) with 16 and 12 filters of size 5x5 respectively, both with ReLU activation and padding.
- Two upsampling layers (upsample1, upsample2) to increase the spatial dimensions.
- Another transposed convolutional layer (upsample3) with 6 filters of size 5x5 and ReLU activation.
- The final transposed convolutional layer (outputs) reconstructs the original image with 3 color channels and ReLU activation.

Layer (type)	Output Shape	Param #
input_layer (InputLayer)	(None, 224, 224, 3)	0
conv2d (Conv2D)	(None, 224, 224, 6)	456
max_pooling2d (MaxPooling2D)	(None, 112, 112, 6)	0
conv2d_1 (Conv2D)	(None, 112, 112, 12)	1,812
max_pooling2d_1 (MaxPooling2D)	(None, 56, 56, 12)	0
conv2d_2 (Conv2D)	(None, 56, 56, 16)	4,816
conv2d_3 (Conv2D)	(None, 56, 56, 3)	1,203
flatten (Flatten)	(None, 9408)	0
dense (Dense)	(None, 9408)	88,519,872
dense_1 (Dense)	(None, 1000)	9,409,000
dense_2 (Dense)	(None, 500)	500,500
dense_3 (Dense)	(None, 1000)	501,000
dense_4 (Dense)	(None, 9408)	9,417,408
reshape (Reshape)	(None, 56, 56, 3)	0
conv2d_transpose (Conv2DTranspose)	(None, 56, 56, 16)	1,216
up_sampling2d (UpSampling2D)	(None, 112, 112, 16)	0
conv2d_transpose_1 (Conv2DTranspose)	(None, 112, 112, 12)	4,812
up_sampling2d_1 (UpSampling2D)	(None, 224, 224, 12)	0
conv2d_transpose_2 (Conv2DTranspose)	(None, 224, 224, 6)	1,806
conv2d_transpose_3 (Conv2DTranspose)	(None, 224, 224, 3)	453

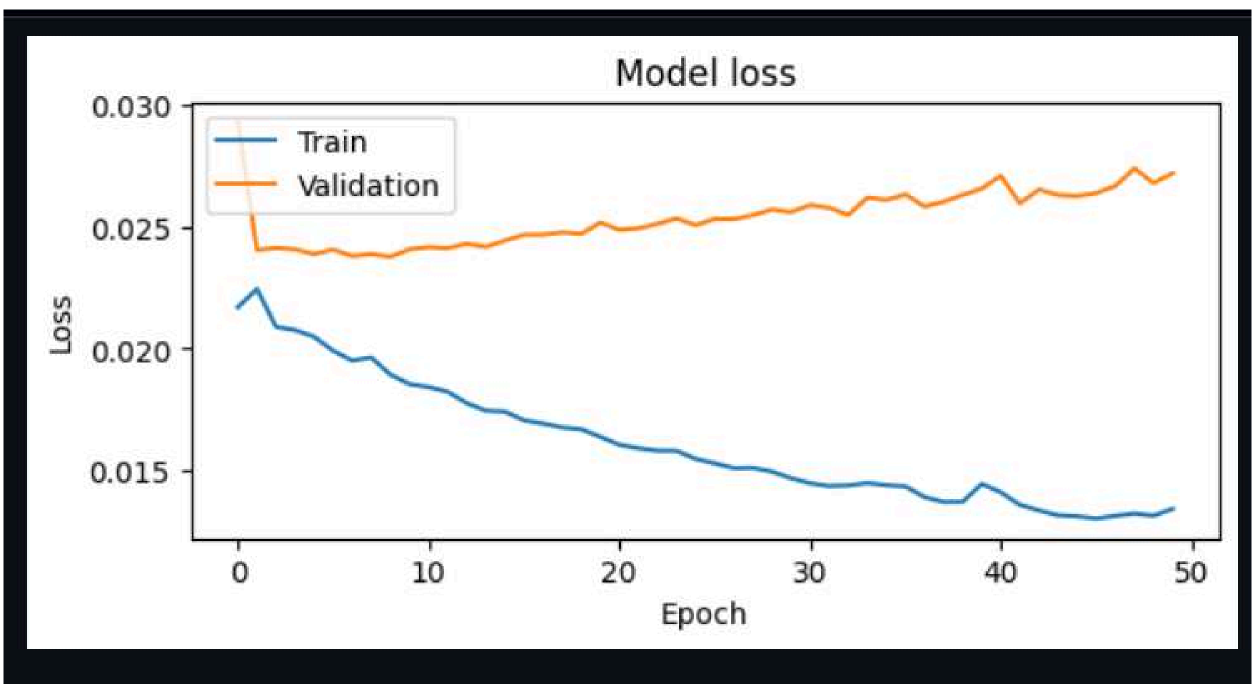
Overview:
The model aims to learn a compressed representation (encoding) of the input images and then reconstruct them using the decoder

part. The use of convolutional layers in both the encoder and decoder helps capture spatial dependencies and patterns in the images effectively. The fully connected layers serve to reduce and then increase the dimensionality of the encoded features. Overall, this architecture constitutes a convolutional autoencoder.

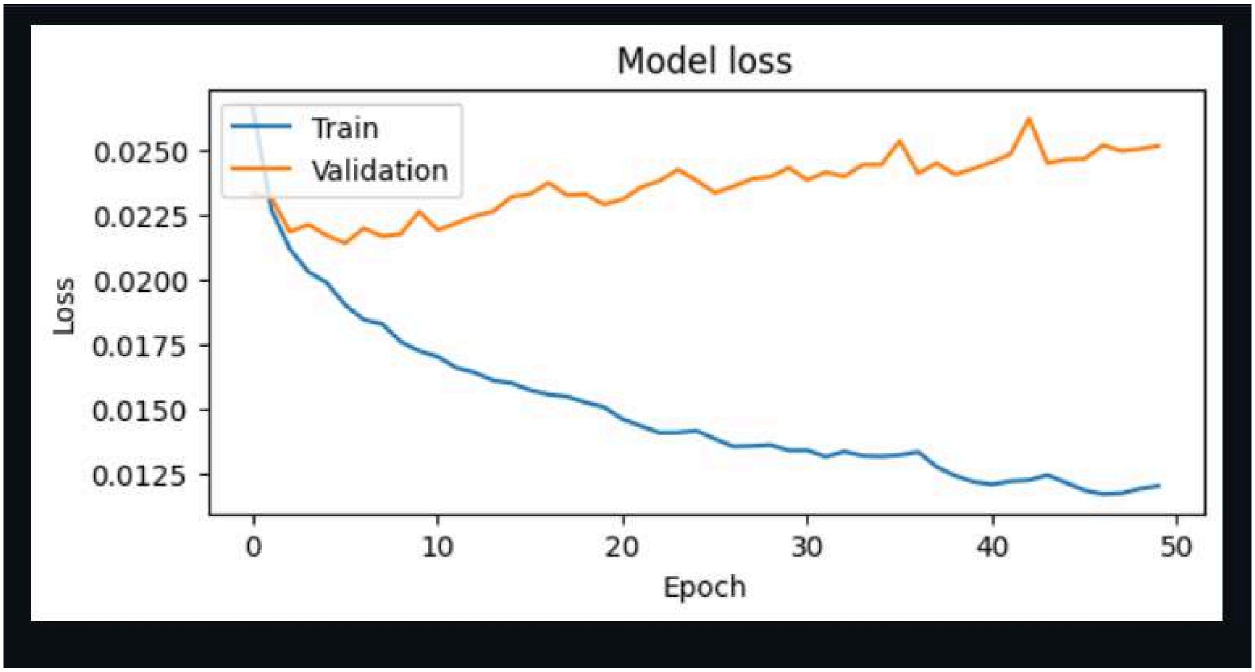
3. Training:

We trained the model with 3 different datasets as presented before, with those models we got good insights.we used our model to fit the data, used batch size 32 and epochs 50 .For Validation we used validation set.For optimizer we used adam, And for loss we used mean squared loss.

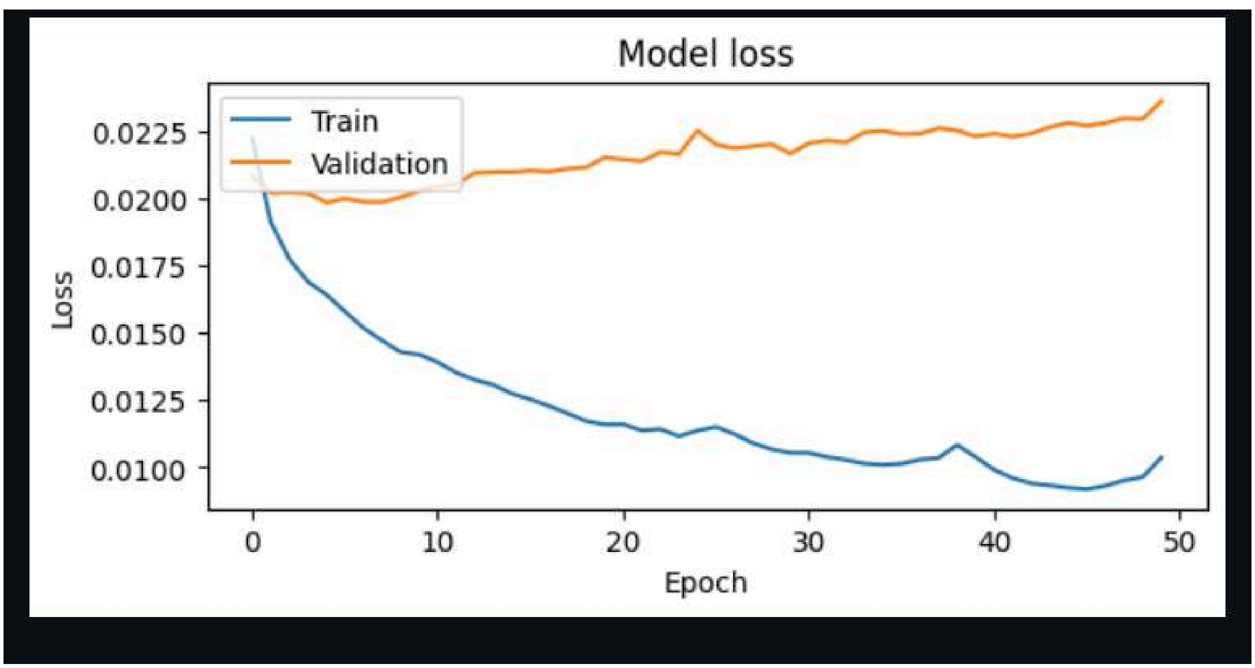
For **Underwater Dark loss**



For **Underwater ImageNet loss**



For **Underwater Scenes loss**

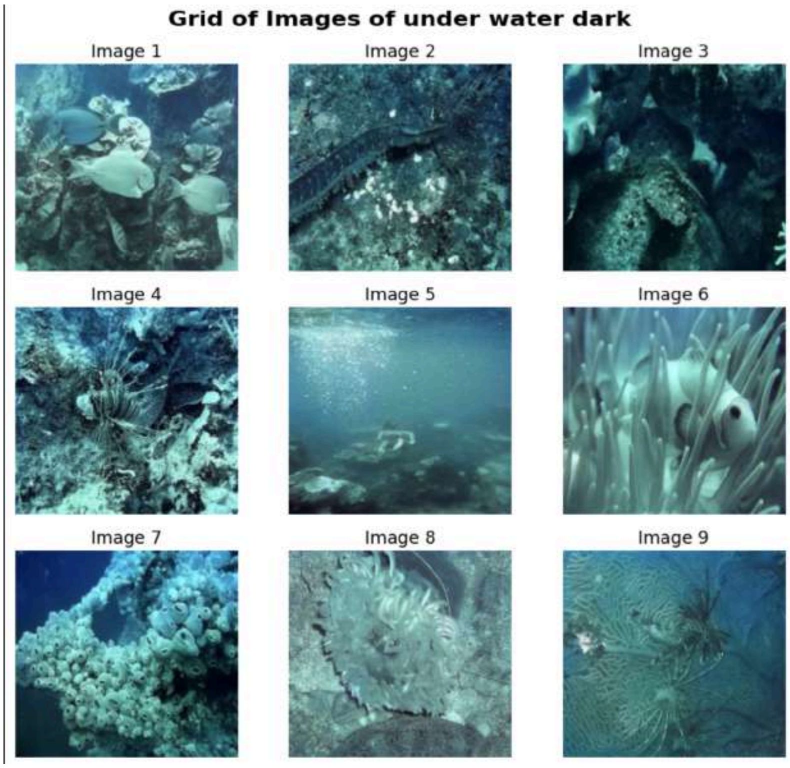


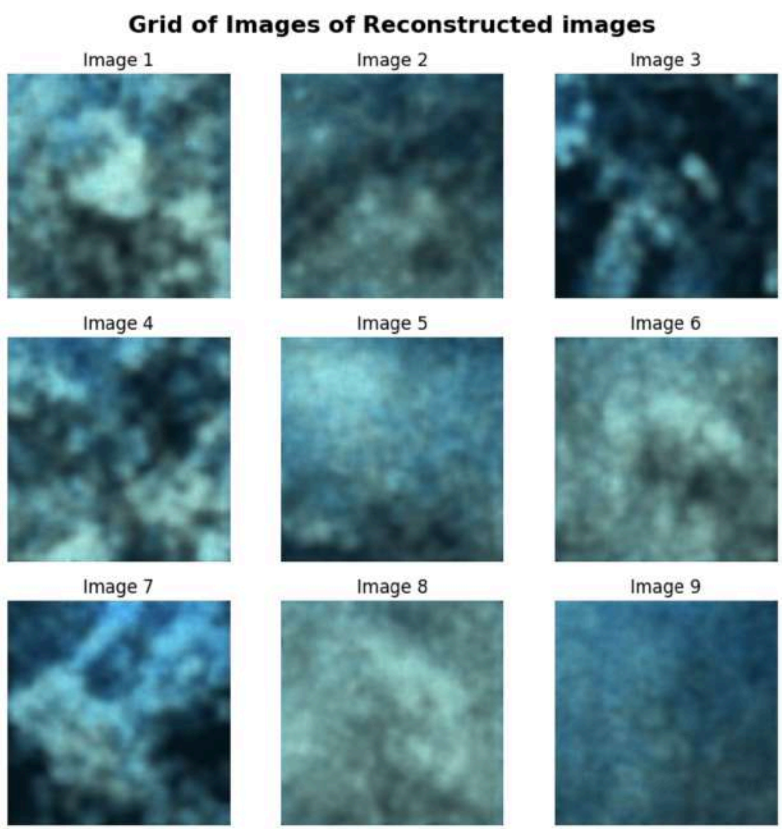
4.Compression: Once the CAE model is trained, we can use the encoder component to compress new images into a lower-dimensional latent space representation. This compressed representation contains the essential information of the original image in a more compact form.Here our latent space dimension is 500.

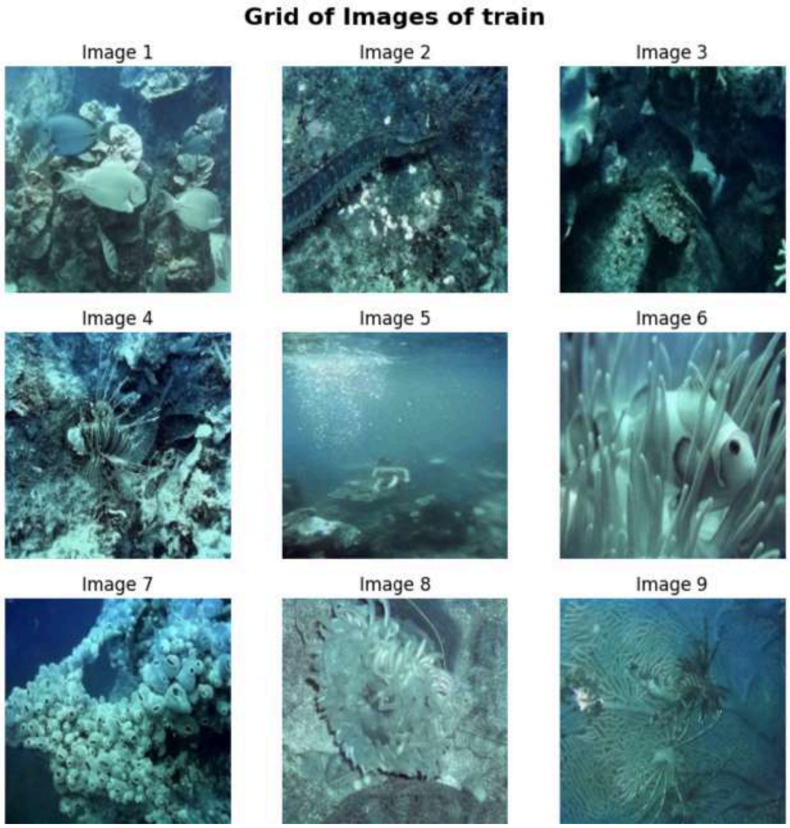
5.Decompression: To reconstruct the compressed image, we use the decoder component of the CAE model. The decoder takes the compressed latent space representation as input and generates a reconstruction of the original image.For 500 latent space dimensions we reconstructed the image.

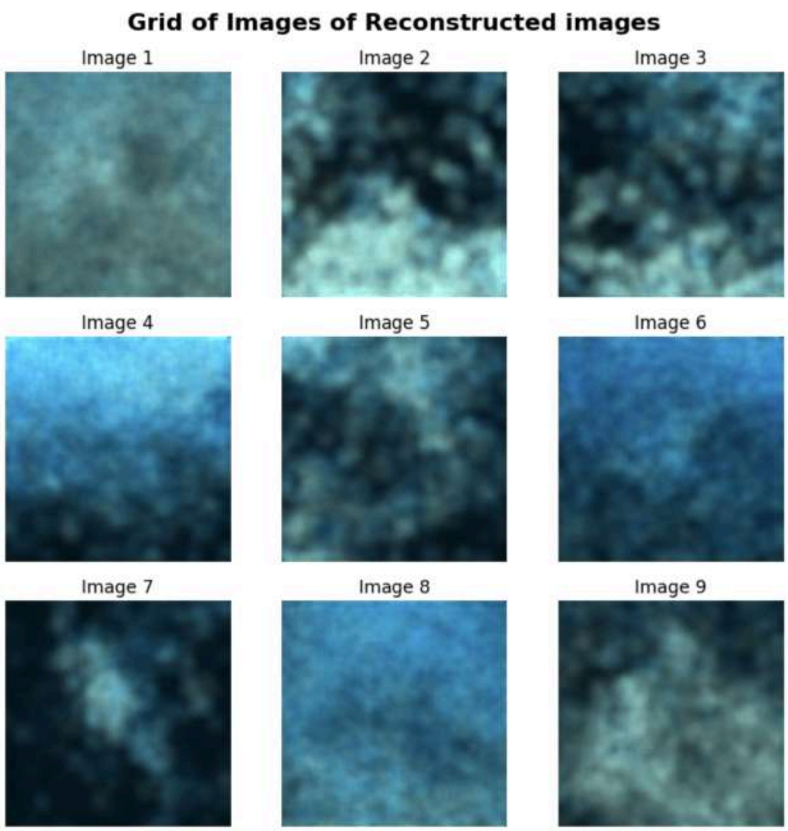
Results and Evaluation:

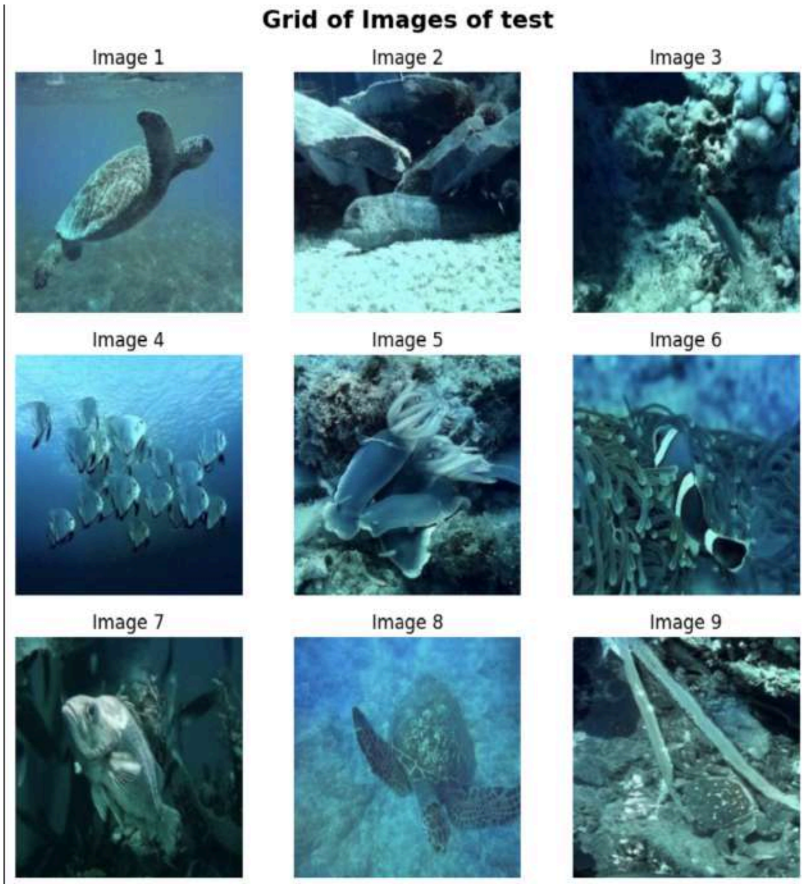
1.Visual Quality: The visual quality of the reconstructed images is assessed subjectively by human observers. We can compare the reconstructed images with their originals and evaluate the level of detail, clarity, and similarity.

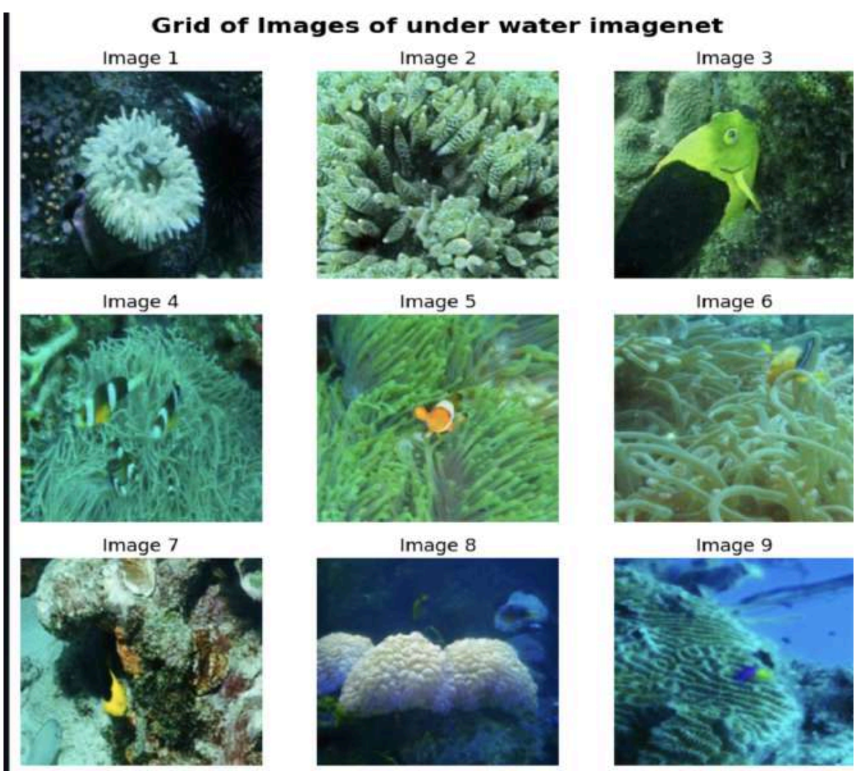

Grid of Images of under water dark
Image 1
Image 2
Image 3
Image 4
Image 5
Image 6
Image 7
Image 8
Image 9

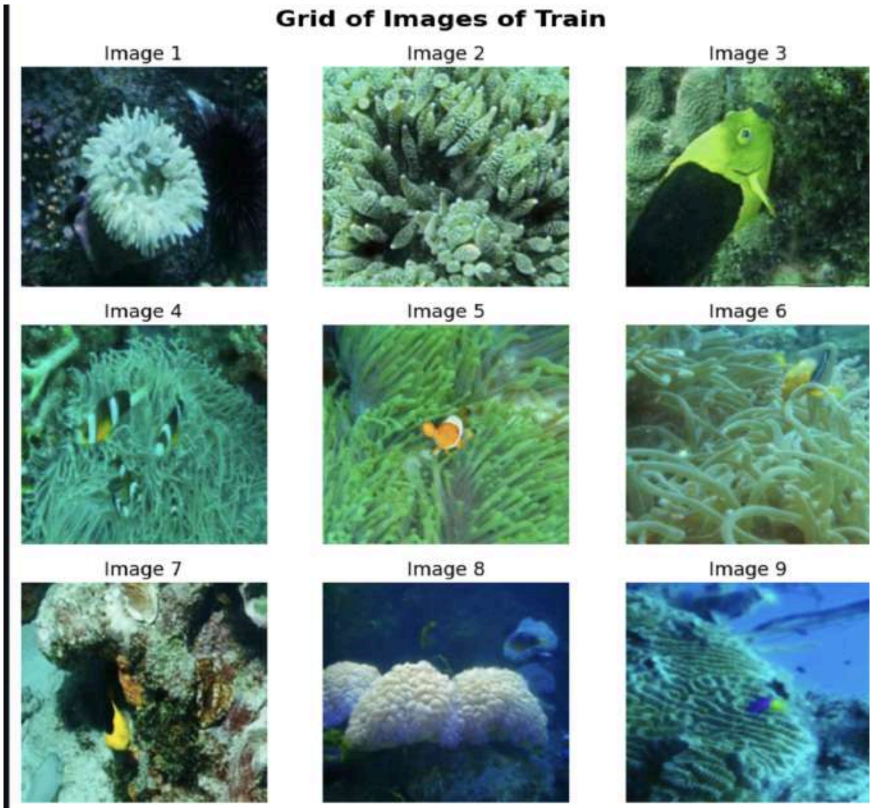

Grid of Images of Reconstructed images
Image 1
Image 2
Image 3
Image 4
Image 5
Image 6
Image 7
Image 8
Image 9

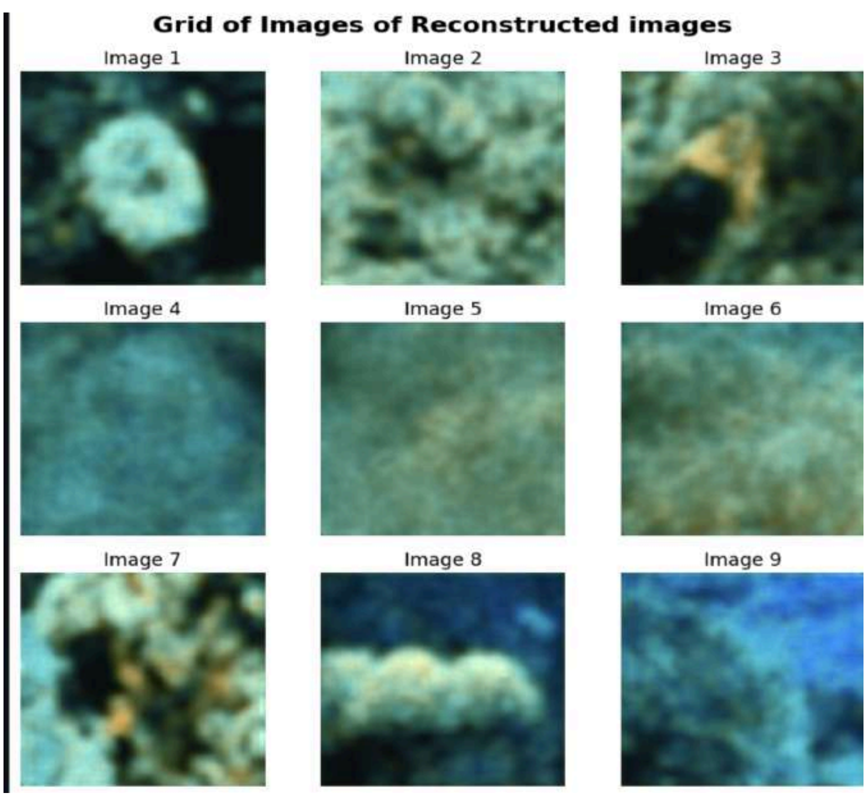

Grid of Images of train
Image 1
Image 2
Image 3
Image 4
Image 5
Image 6
Image 7
Image 8
Image 9

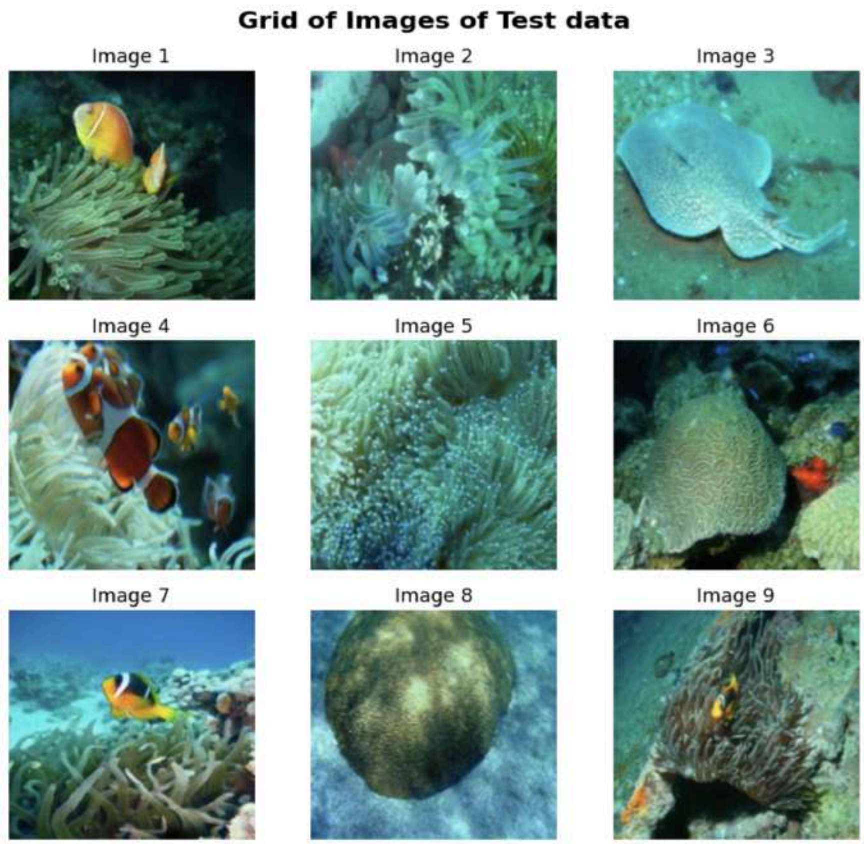

Grid of Images of Reconstructed images
Image 1
Image 2
Image 3
Image 4
Image 5
Image 6
Image 7
Image 8
Image 9

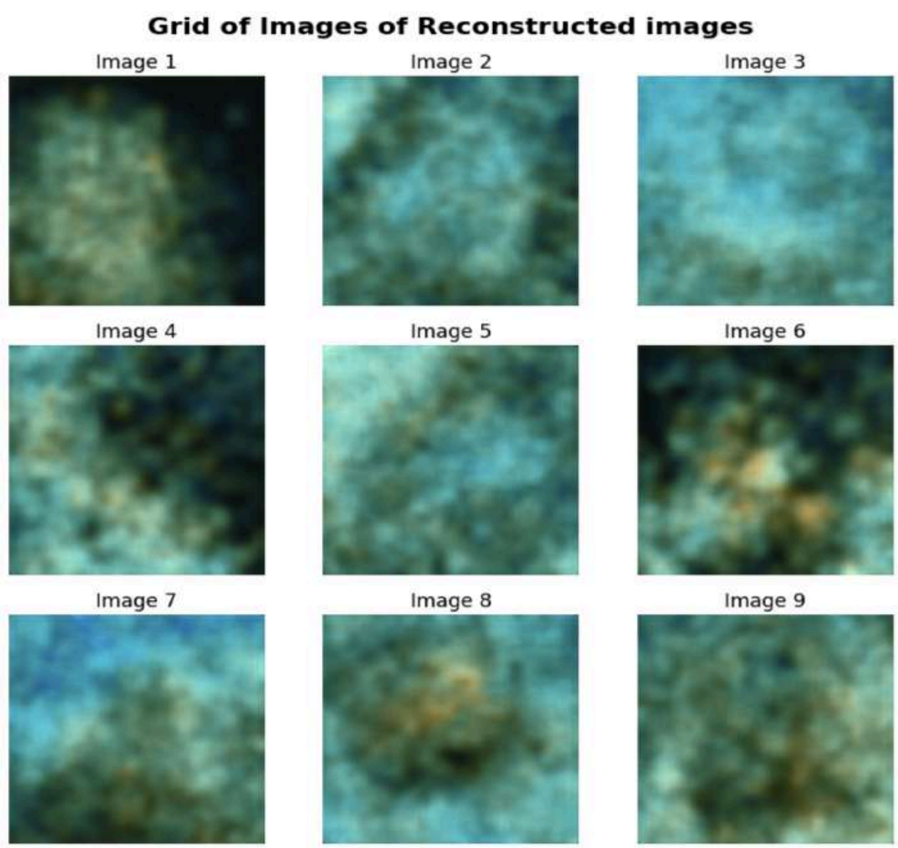

Grid of Images of test
Image 1
Image 2
Image 3
Image 4
Image 5
Image 6
Image 7
Image 8
Image 9


Grid of Images of under water imagenet
Image 1
Image 2
Image 3
Image 4
Image 5
Image 6
Image 7
Image 8
Image 9

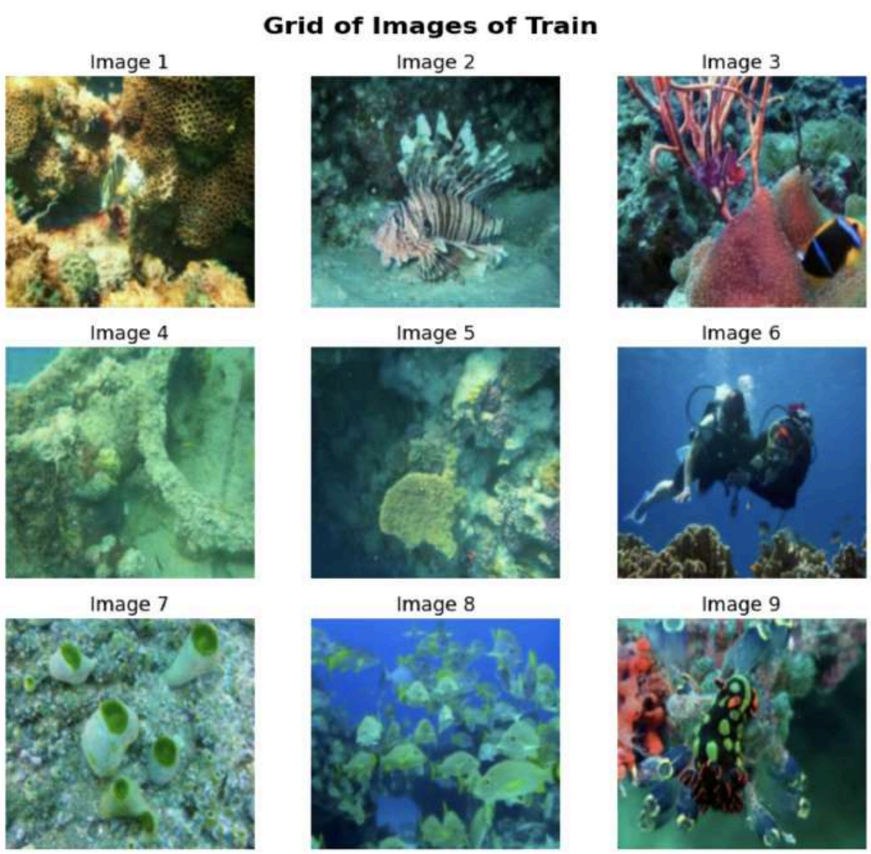

Grid of Images of Train
Image 1
Image 2
Image 3
Image 4
Image 5
Image 6
Image 7
Image 8
Image 9

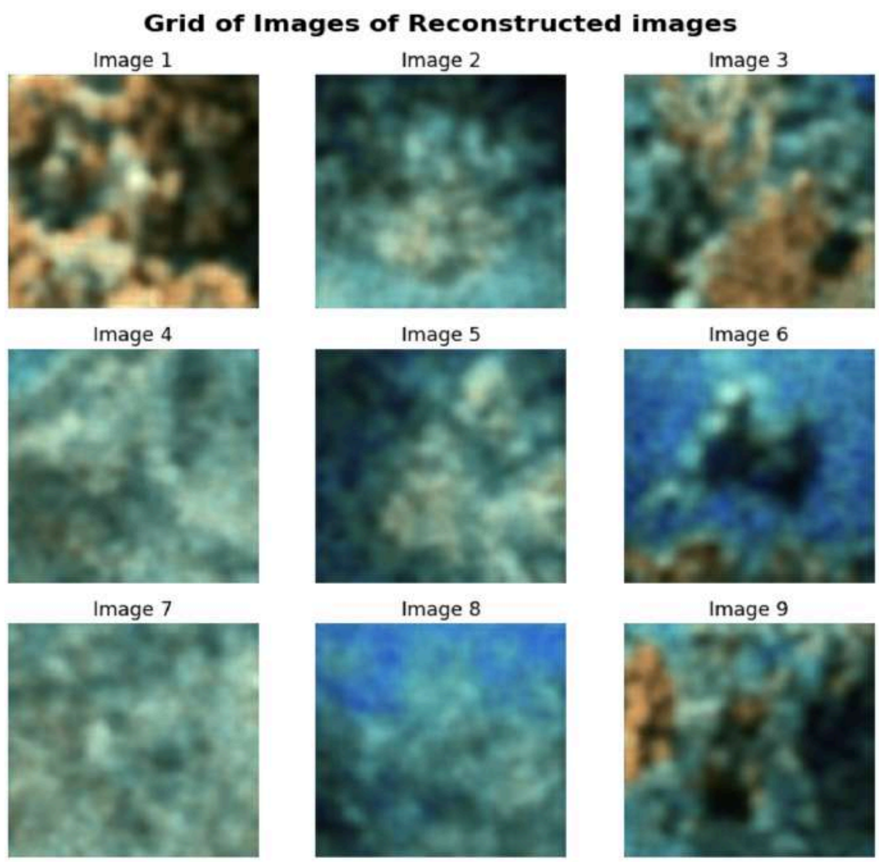

Grid of Images of Reconstructed images
Image 1
Image 2
Image 3
Image 4
Image 5
Image 6
Image 7
Image 8
Image 9

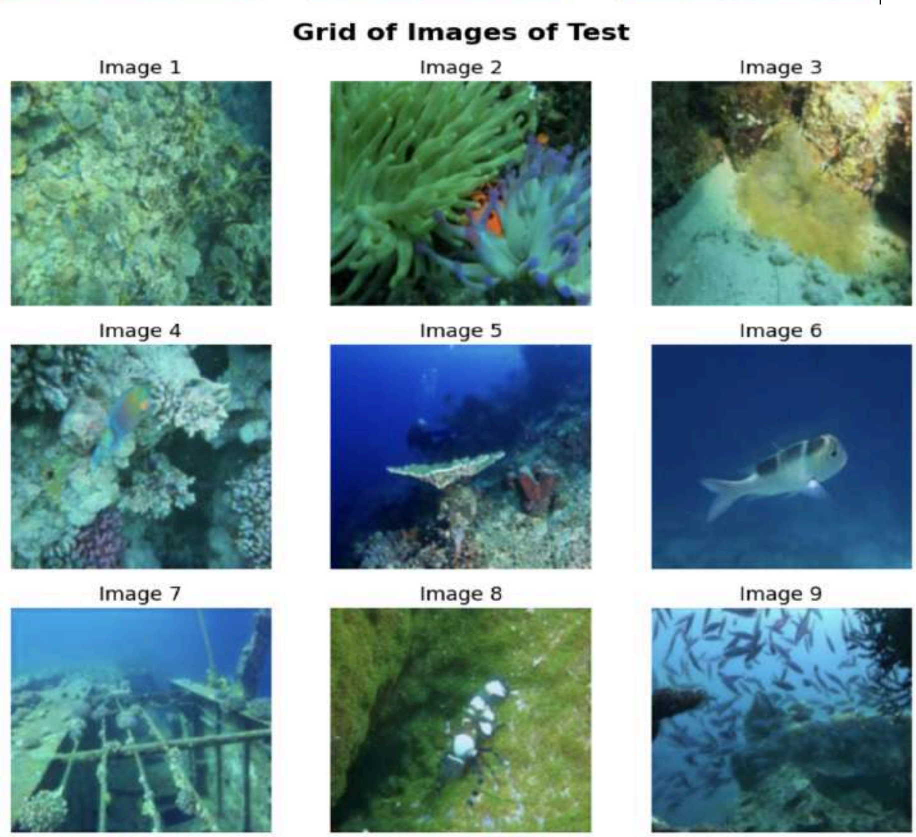

Grid of Images of Test data
Image 1
Image 2
Image 3
Image 4
Image 5
Image 6
Image 7
Image 8
Image 9

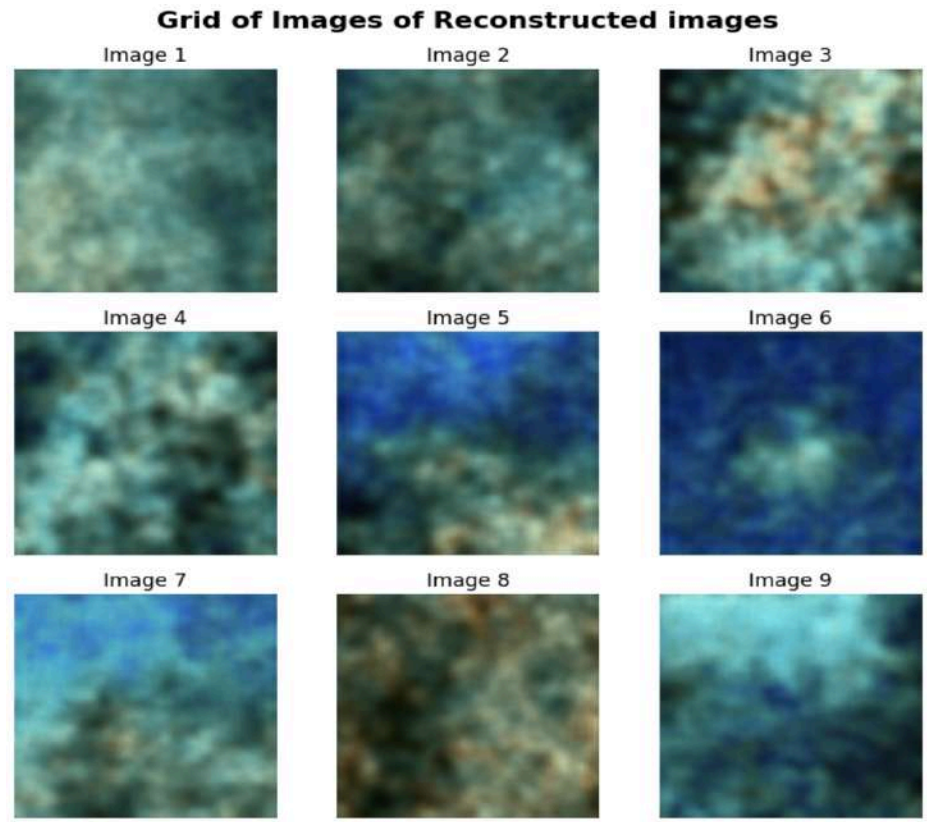

Grid of Images of Reconstructed images
Image 1
Image 2
Image 3
Image 4
Image 5
Image 6
Image 7
Image 8
Image 9


Grid of Images of under water scenes
Image 1
Image 2
Image 3
Image 4
Image 5
Image 6
Image 7
Image 8
Image 9


Grid of Images of Train
Image 1
Image 2
Image 3
Image 4
Image 5
Image 6
Image 7
Image 8
Image 9


Grid of Images of Reconstructed images
Image 1
Image 2
Image 3
Image 4
Image 5
Image 6
Image 7
Image 8
Image 9


Grid of Images of Test
Image 1
Image 2
Image 3
Image 4
Image 5
Image 6
Image 7
Image 8
Image 9



2.Quantitative Metrics: Quantitative metrics such as Peak Signal-to-Noise Ratio (PSNR) and Mean Squared Error(MSE) can be used to objectively evaluate the quality of the reconstructed images. Higher PSNR and lower MSE values indicate better reconstruction quality.

PSNR For different Datasets

PSNR	Train	Test

Underwater Dark	67.4	64.2
Underwater ImageNet	67.8	64.5
Underwater Scenes	68.1	64.9

MSE For different Datasets

MSE	Train	Test
Underwater Dark	0.012	0.027
Underwater ImageNet	0.011	0.025
Underwater Scenes	0.010	0.022

Conclusion:

Convolutional Autoencoders offer an effective approach to image compression by leveraging deep learning techniques. By learning compact representations of images in a lower-dimensional latent space, CAEs can achieve significant compression while preserving visual quality. Future research may focus on improving the efficiency and effectiveness of CAEs for real-world applications, such as lossy and lossless image compression, video compression, and bandwidth-constrained environments.