Mukesh Dharanibalan

Machine Learning – URoP Tasks

26th December 2023

Evaluation of Machine Learning Models and Prediction Accuracy:

Convolutional Auto Encoders, Principal Component Analysis, Long-Short-Term-Memory, and Gated Recurrent Units.

This short report-draft is used to explain figures produced by included code-files, the resultant outputs, and an evaluation of these results and their consequences regarding the general problem. Given that the problem is fundamentally dependent on compressing and decompressing data - to minimize computational burden and efficiency of data processing – it important to evaluate different methods to do so to determine the most appropriate. Here, Principal-Component based data-compression, and a Convolutional Auto-encoder is considered, and their performance compared.

Using the scikit-learn, PyTorch, matplotlib and numpy libraries, and by choosing the same videos from the provided dataset for consistency between methods, the reconstruction accuracy can be measured. Videos 45, and 47 were chosen based on their performance in the auto-encoder. The CAE architecture consisted of 6 2-D Convolutional Layers (with ReLU activation layers), and 5 Max-Pooling2d layers for the encoder. The decoder was symmetric with MaxUnPooling2d layers in-stead to maintain utmost accuracy across compression and decompression. This meant that the spatial dimensions of the dataset were reduced from (128 x 128) to (4 x 4) when compressed, with 128 feature maps per 16 frames per video.

This model was trained using the first 33 videos in the dataset, with 8 others used for validation, over 100 epochs. Improvements to this method could include use of an early stopping

conditions, or a threshold to terminate further iterations. The performance of the model across different videos in the test dataset – which consisted of the remaining 8 videos – is shown in Fig. 1.

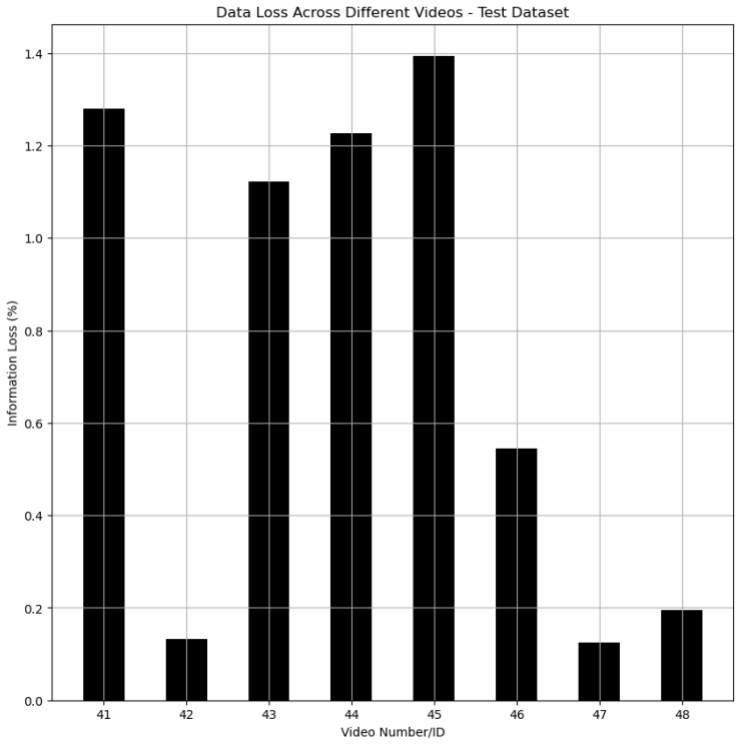


Fig 1. Percentage data loss when using optimised CAE model on the testing videos dataset. The average loss is ~0.75%. Binary Cross Entropy was the preferred loss function as videos had binary channels.

The Principal Components of a dataset are essentially linear combinations of all the features/variables (with varying weights) present in an input. The sci-kit learn library, and the fit method, these weights are optimised like a 'line of best-fit' where n-Principal Components are included.

Principal Component analysis is done by analysing the dependence of the Cumulative Variance explained on the number of principal components included during data compression. This is an effective measure of the amount of preserved after data compression and decompression.

This dependency is shown on a Scree-plot. Which facilitates visual determination of the smallest choice of principal components that maximise the cumulative variance explained. These plots are shown for videos 45, and 47, the reasons for which are seen later when considering reconstructions of these images.

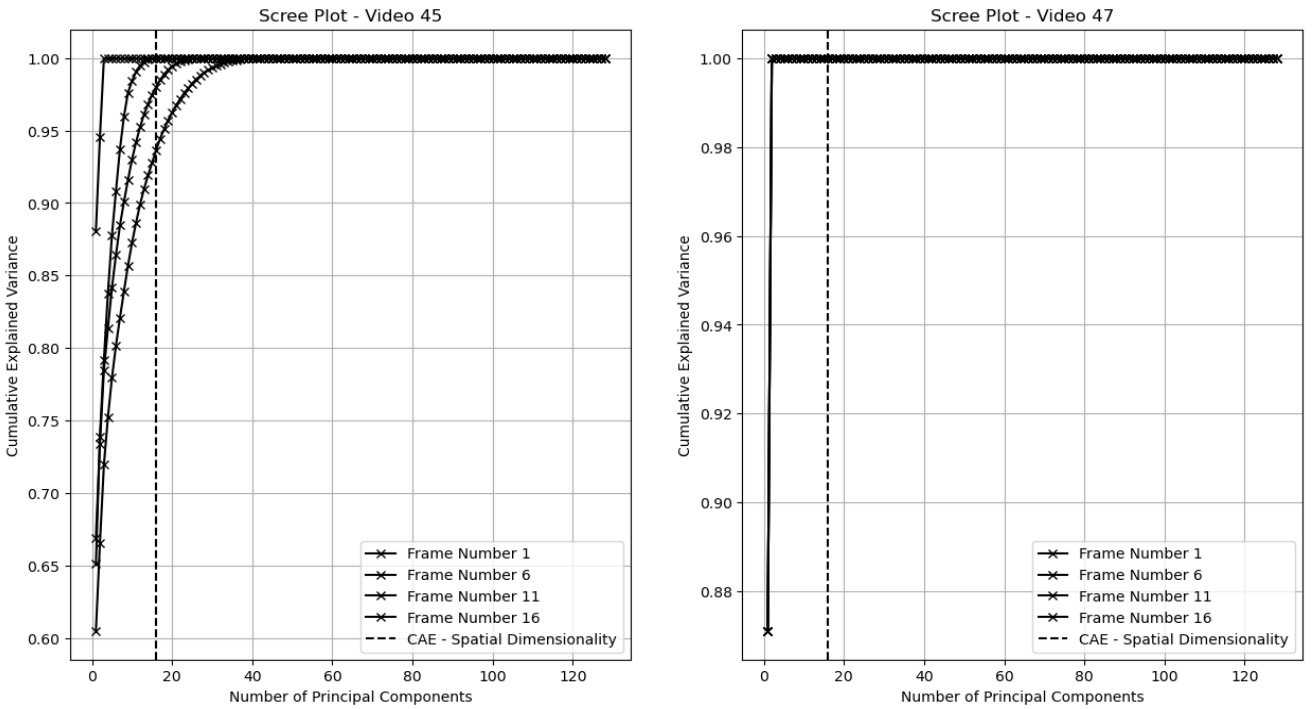


Fig. 2. Scree plot for Videos 45 and 47, vertical line shows the spatial dimensionality of the CAE's compressed space data. Significant data-loss is seen in the later frames for video 45 as the intersection between the threshold and cumulative variance explained is ~94% in the worst case.

This clearly demonstrates the limitations of PCA in matching the dimensionality of the CAE. Not to mention, the CAE can be compressed further to have a dimensionality of 2 x 2 if required and only improves in performance as the dataset grows. Furthermore, following a short period of unsupervised training, the CAE can work generally, when applied to a diverse set of datasets. PCA is case specific, and the fitting must be done on a video-to-video basis limiting is applicability and computational efficiency. It is however important to note that the

MaxUnPooling2d is symmetric with the MaxPooling layers, and requires the indices from these to function, a limitation observed in the prediction tasks.

The reconstruction can also be compared visually, seen in Fig. 3 and 4, where the limitations of each method are clearly observed, showing the data-loss in the principal-Component compression, and the inherently lossy nature of a CAE.

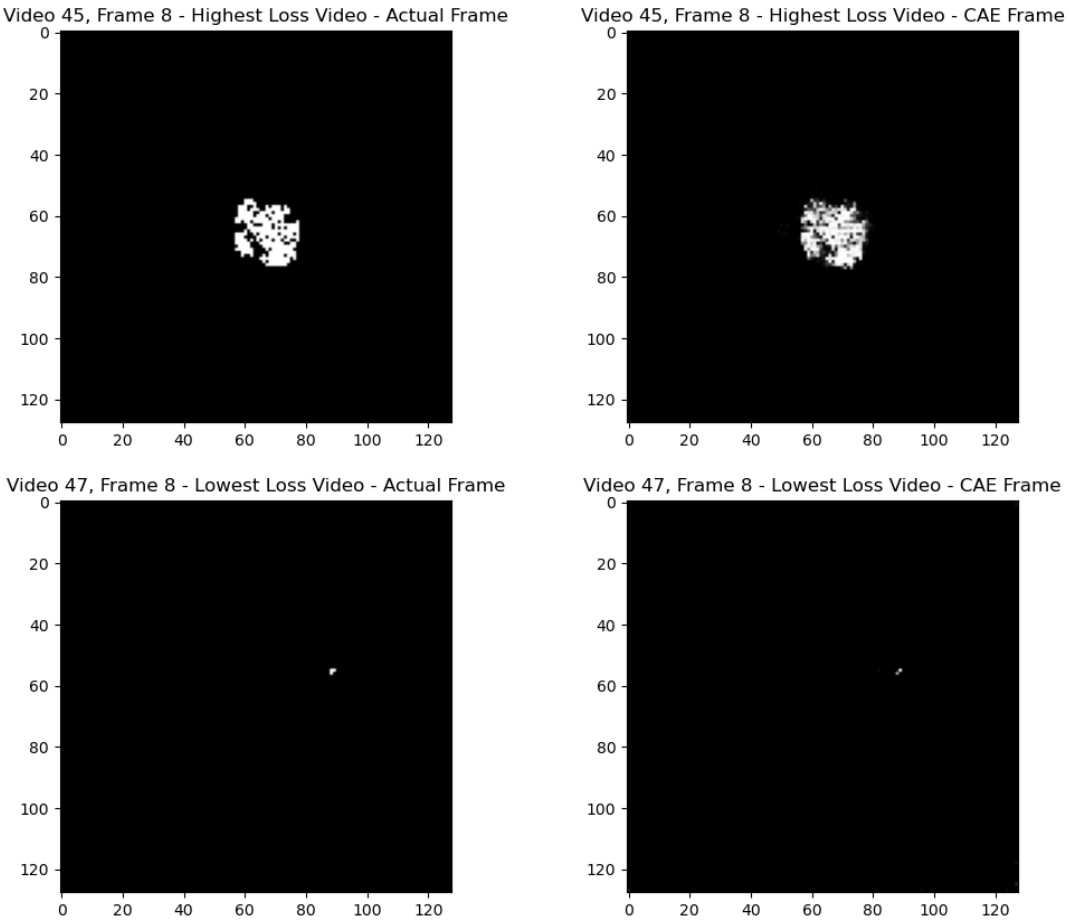


Fig. 3. Reconstruction of Videos 45 and 47 using CAE compression and decompression. An arbitrary frame is chosen; however, the behaviour is observed across all frames. Minor artifacts are observed, displaying clear limitations in the pooling layers.

Edge artifacts, loss of brightness and poor preservation of the binary-features of the dataset are clearly seen in the Principal Component reconstruction. Evident is the reasons for the extremely low errors for both methods even at such low dimensionality for video 47. All frames consist of a

very small number of features, reducing any amount of possible data-loss, and easing data-compression and increasing preservation.

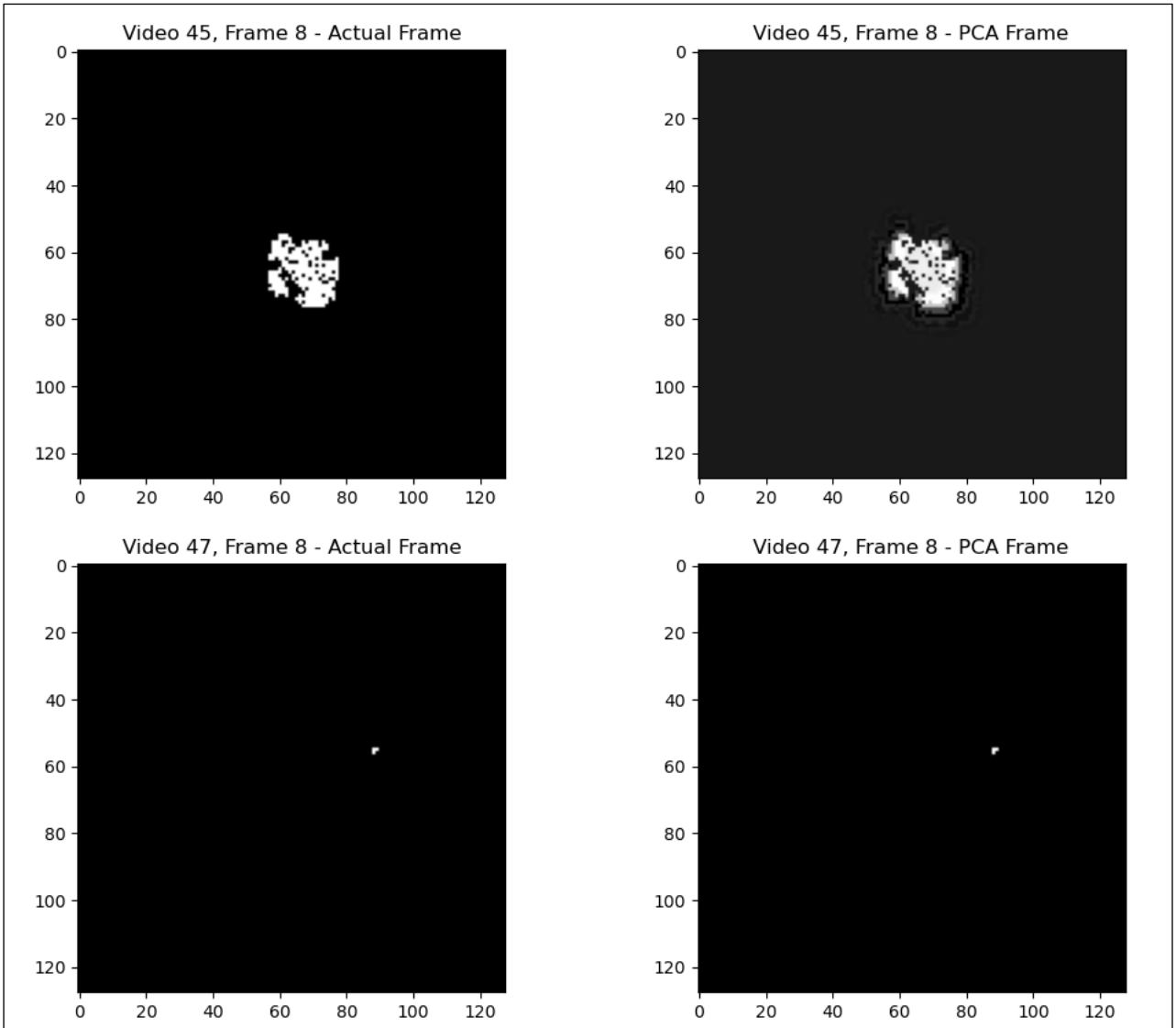


Fig. 4. Principal Component Reconstructing with 16 principal components. Clear fading around the edges is observed. Likely due to the number of components barely covering the range of spaces occupied by the field, leading to feature loss and poor data preservation. The lack of generality is also a significant limitation in application. Noise in the background is also clearly observed in Video 45.

Having established the better option – the CAE – due to its ubiquity and accuracy for now, a sequence-to-sequence prediction model build was the next objective. The same videos and video-splits were used for training, validation, and testing of the model. An LSTM, and GRU algorithm-

based predictor were trained, where different loss functions were used: Binary Cross Entropy (BCE), Mean-Squared Error (MSE), Structural Similarity Index Metrix (SSIM) + BCE and SSIM + MSE. Out of which the best option could be used in terms of reconstruction accuracy. Processed in batches of 11 videos (when training) 12 frames from each video were used as the basis and aimed to predict the next 4 frames. The accuracy of which was compared to the remaining 4 out of 16 target frames.

The basis range and target range can be appropriately amended as required. However, this split offered a good balance between adequate samples per video when making a prediction – increasing accuracy, and number of target frames adequately evaluating multi-step prediction accuracy. Fig. 5. Displays the performance of different algorithms and loss functions clearly, allowing us to narrow down methods to thoroughly evaluate here.

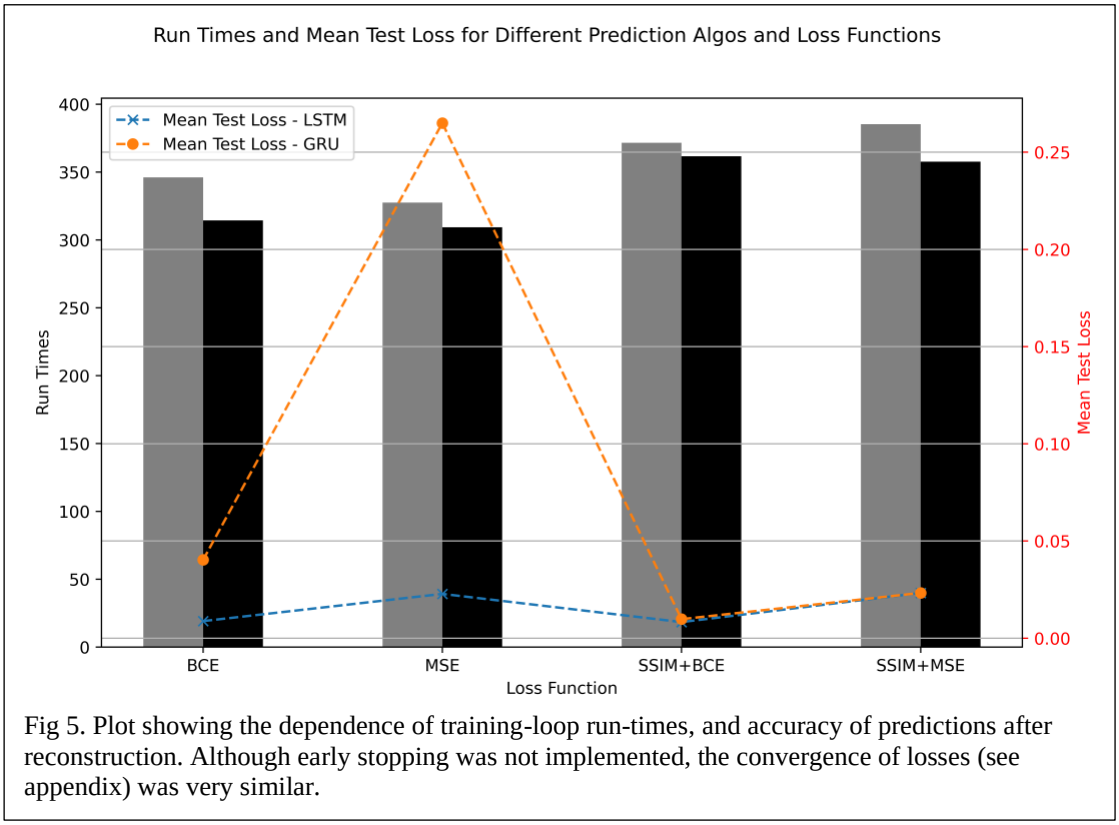


Fig 5. Plot showing the dependence of training-loop run-times, and accuracy of predictions after reconstruction. Although early stopping was not implemented, the convergence of losses (see appendix) was very similar.

The limited number of adjustable weight parameters, fewer layers (2 rather than 3) and worse feature preservation/reading means the GRU is a clearly inferior prediction method. The longer run-time likely arrives from the model attempting to compensate by fine-tuning up sampling (ConvTranspose2d) layers in the decoder, and possible less-optimised hypermeter choice. Up-Sampling techniques had to be used as MaxPooling indices are not readily available for the predicted dataset. The effects of this are clearly seen by smearing in the reconstructed dataset, although general features are maintained.

Thus, focusing on the LSTM dataset, the reconstruction accuracy can be visually evaluated, seen in the images below. Clearly seen, in Fig 7 and 8, any use of an MSE loss function results in no output. This is because of a lack of compatibility of the method with the binary scale, poor gradient behaviour, and outlier sensitivity, resulting in significant loss of information, when parameters are tuned to suit it.

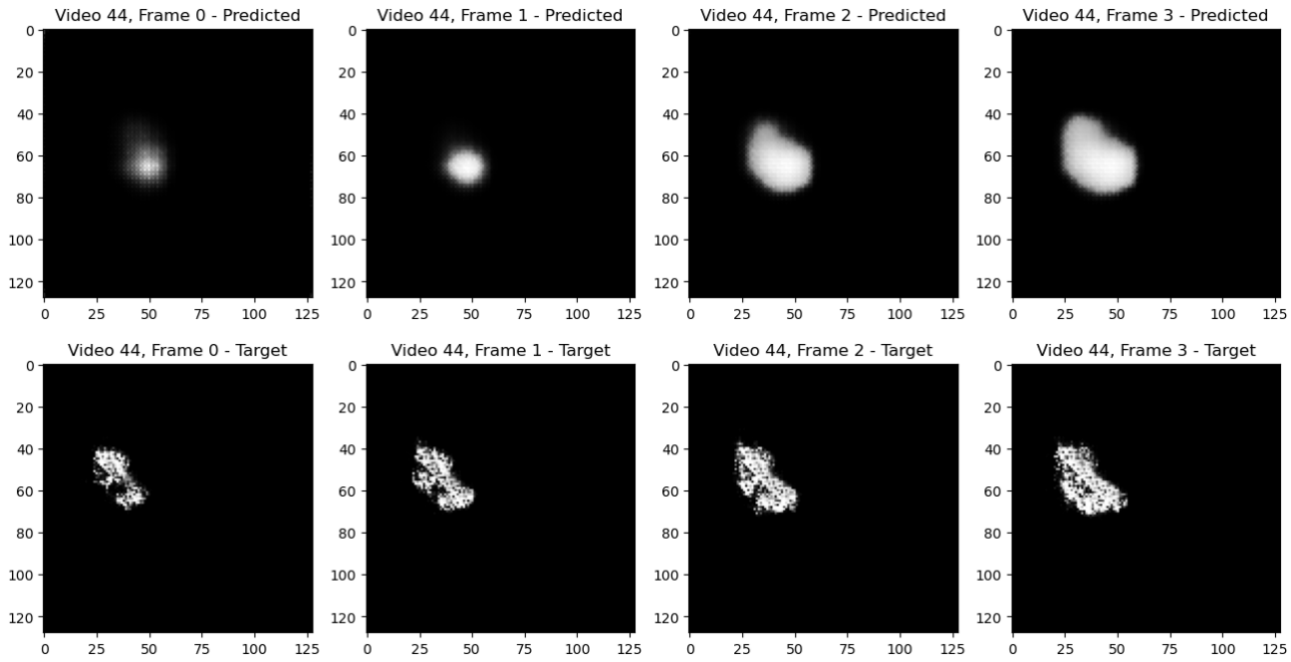


Fig. 6. Images showing videos, target frame number and expected output. The target frames were compressed and decompressed to reduce strain on model parameters and prevent overfitting. This however does reduce general accuracy. Smearing and loss of perfect binary pixels is clearly scene, a result of windowed-up sampling. BCE LOSS FUNCTION USED.

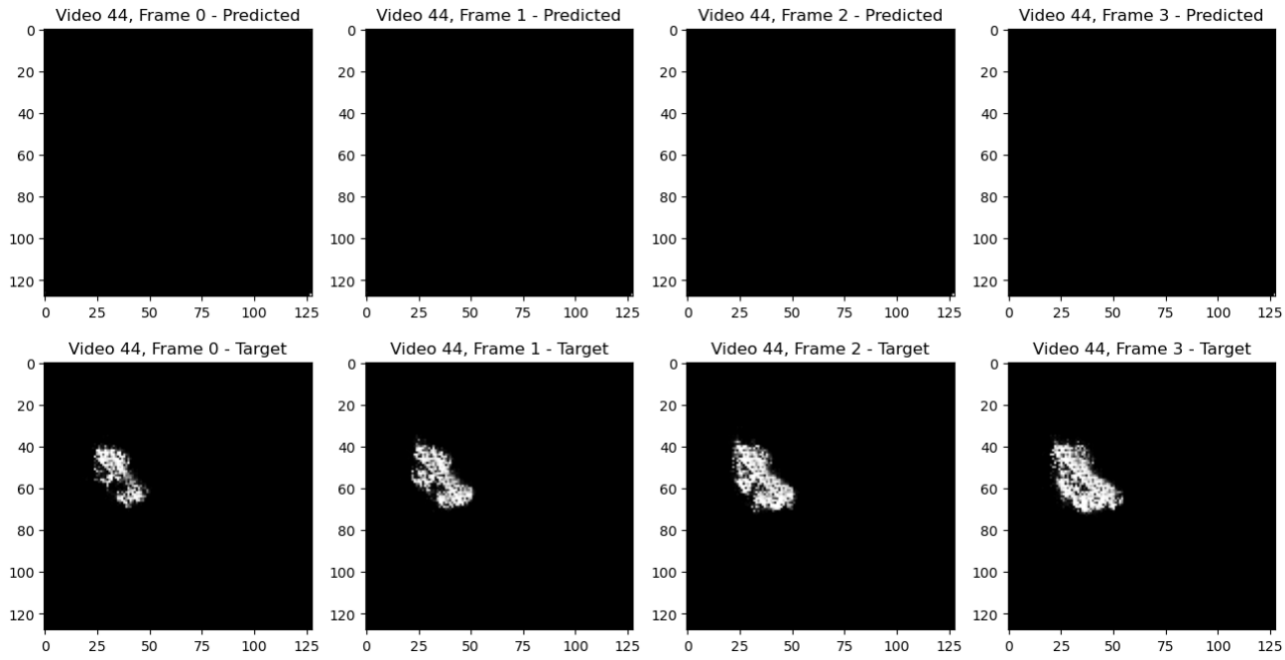


Fig. 7. Same video, now reconstructing using MSE loss function. Complete loss of information is seen due to scale-mismatch and gradient behaviour when calculated by MSE.

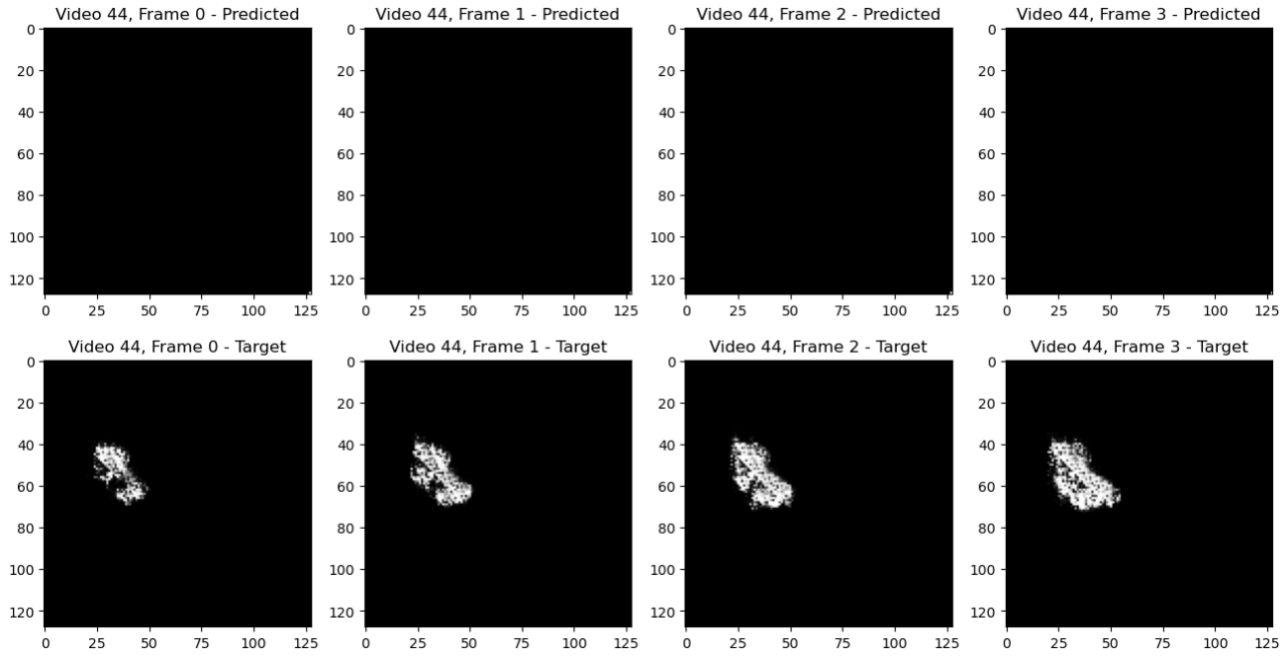


Fig. 8. Same video choice, reconstruction/prediction accuracy when SSIM+MSE loss functions are used.

The shape of video 44 means that it is unique compared to the entire dataset that is used for training/testing which means that it produces the greatest loss values, and therefore has the lowest accuracy of predictions and reconstructions. By far the best, with a mean loss of only ~0.83% is the SSIM + BCE loss function with an LSTM predictor, producing the results seen in Fig, 9. Most of the error is likely due to decompression induced inaccuracies rather than prediction inaccuracy.

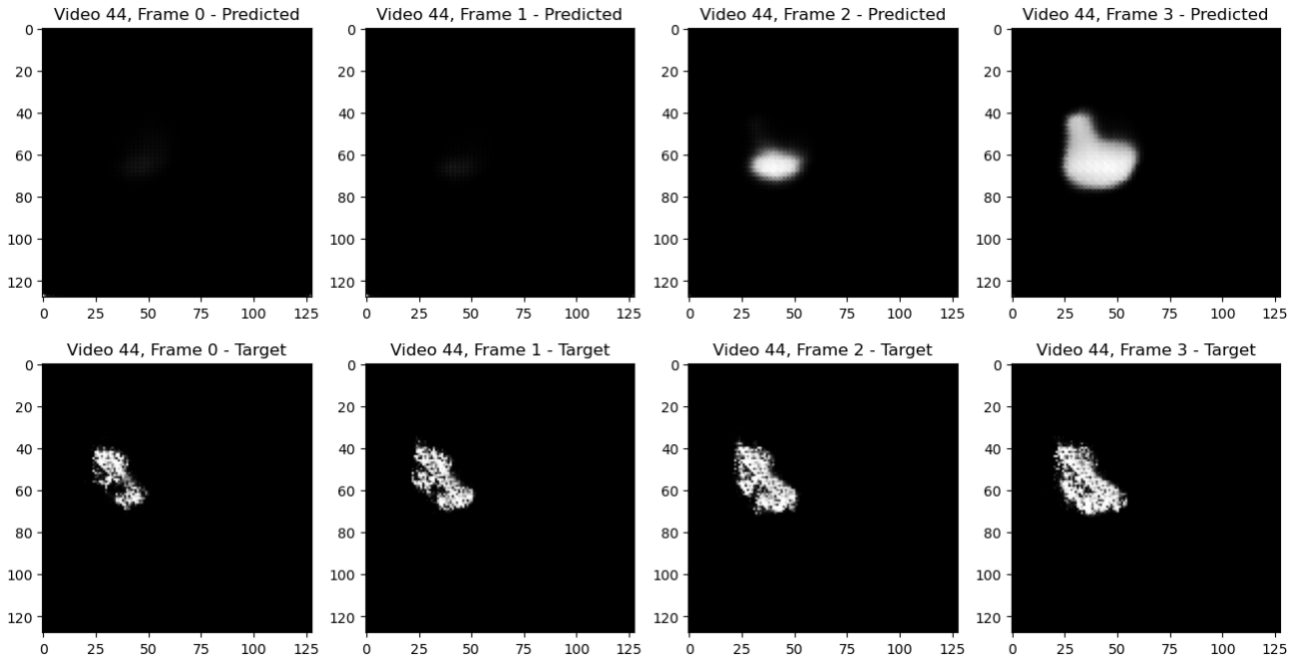
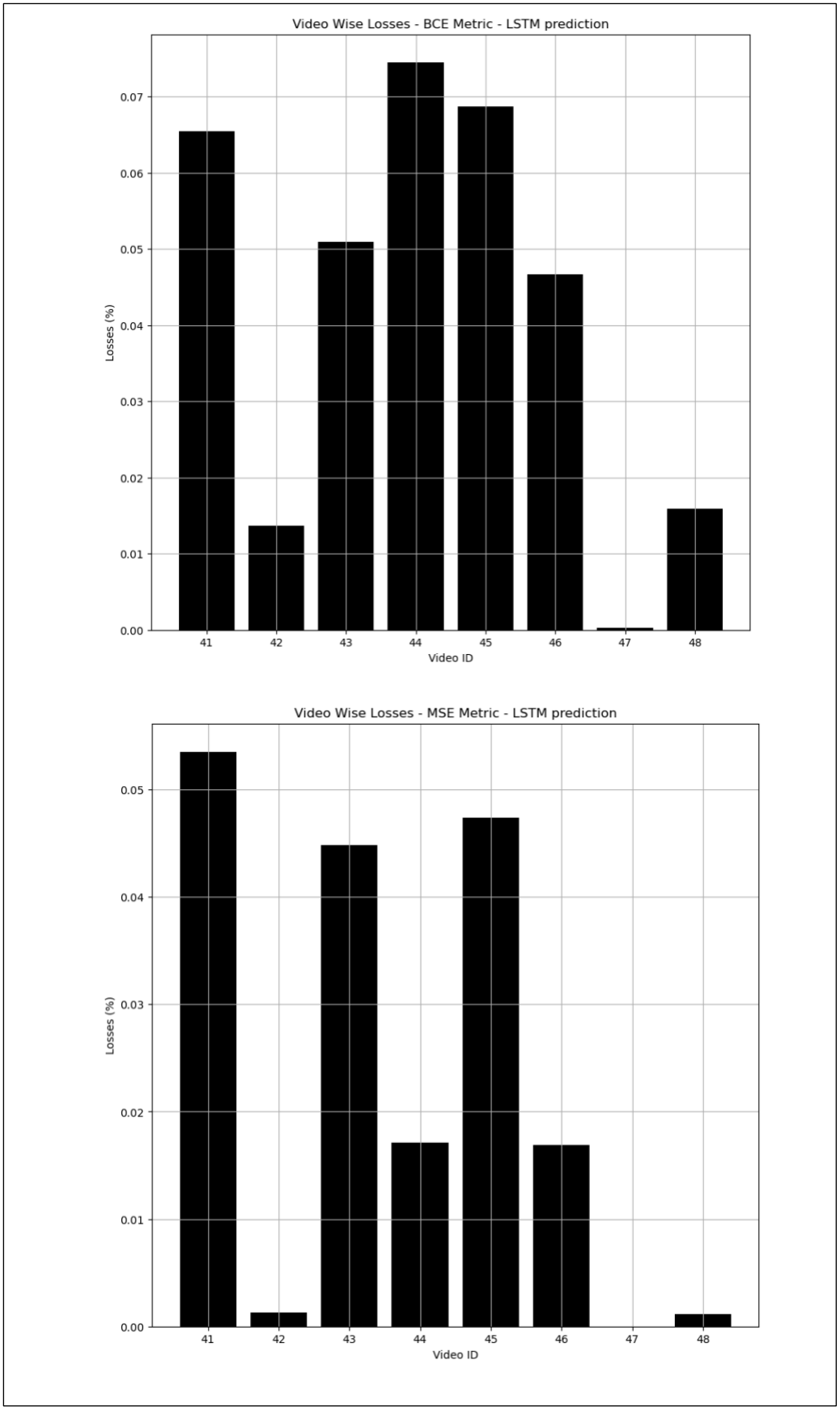
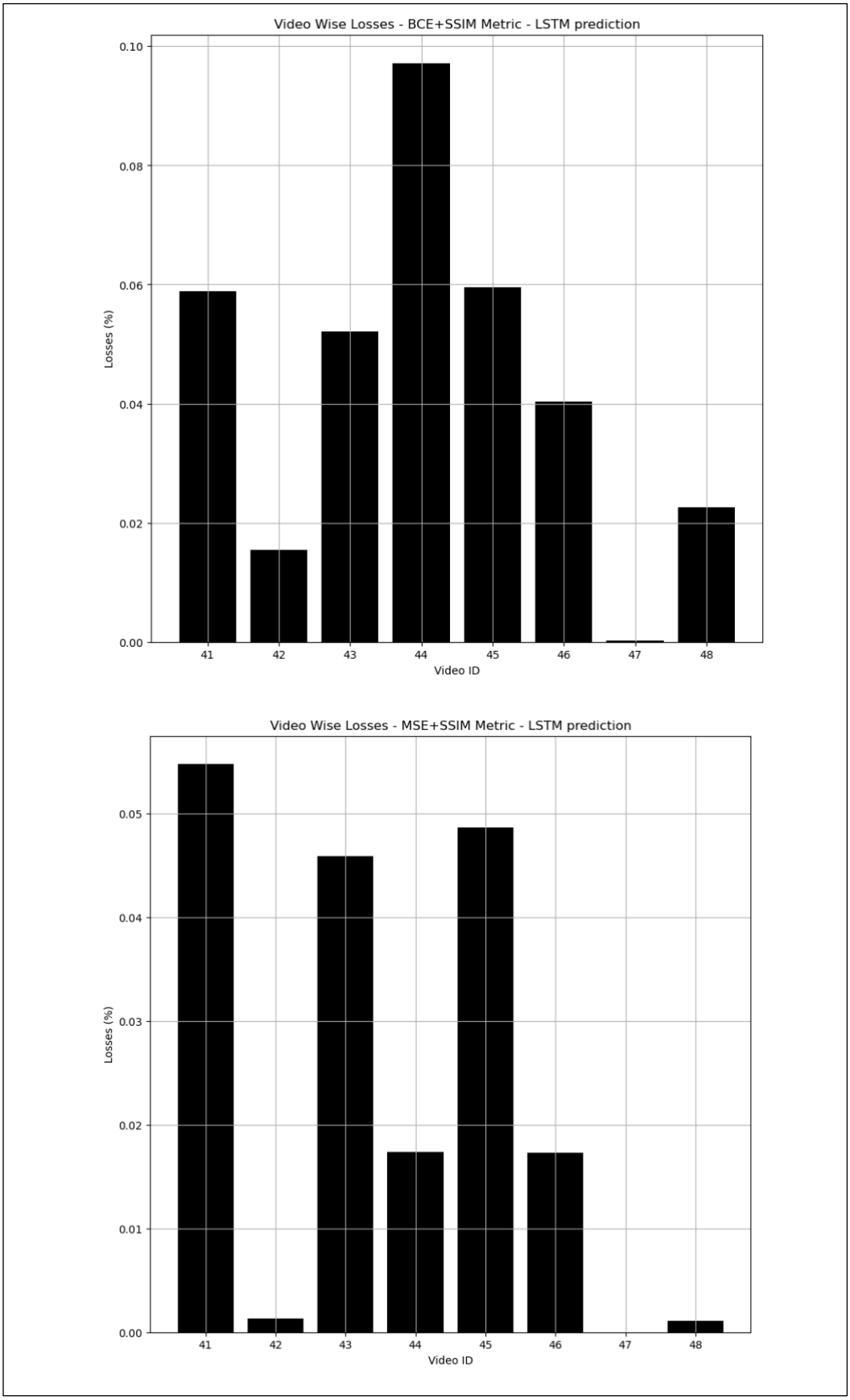


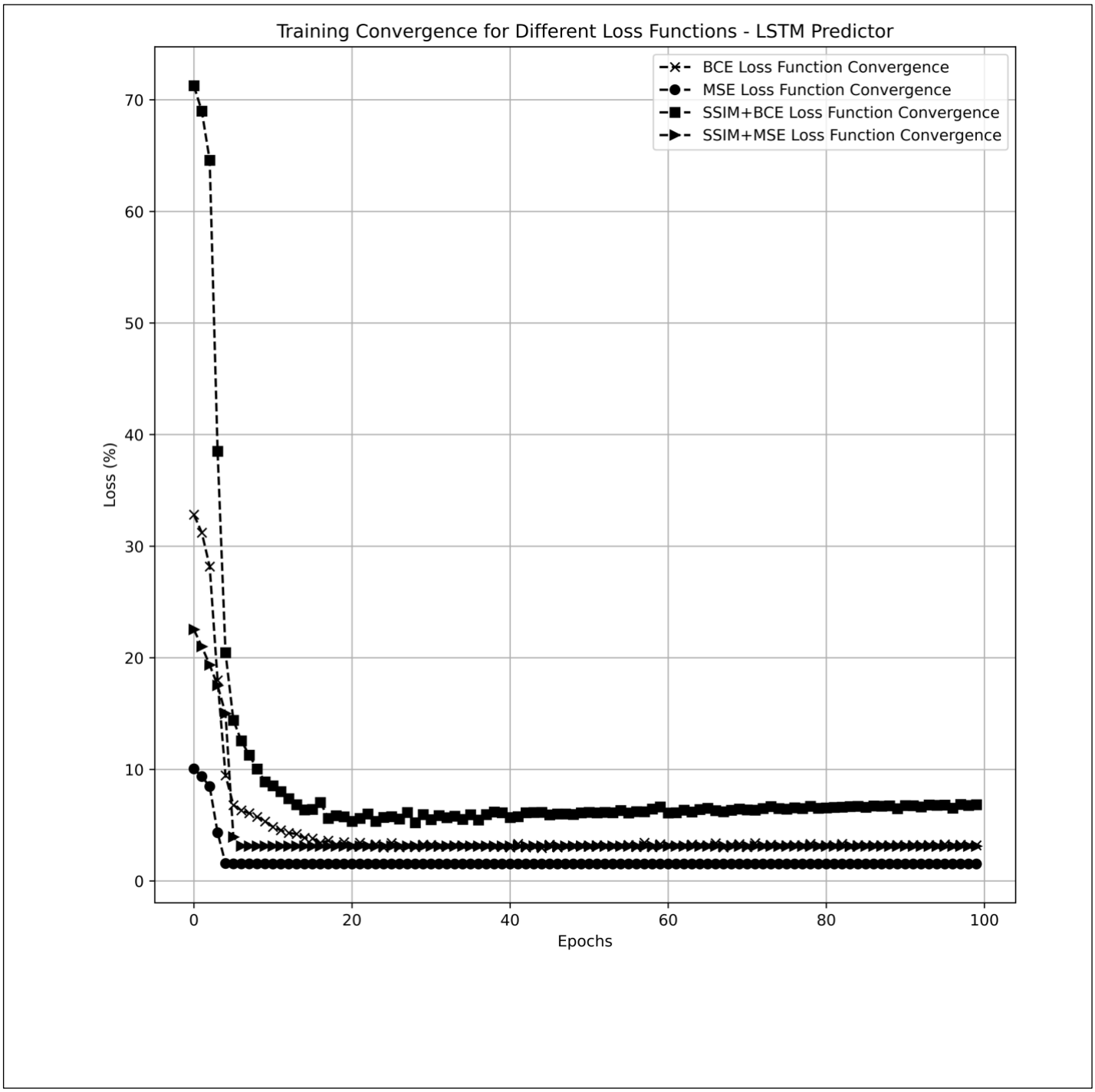
Fig. 9. Predicted and target images produced by an LSTM layer. SSIM + BCE Loss function implementation was used here. Clearly seen however, is the vast inaccuracy for this video compared to the BCE method alone. This is because the SSIM method is highly optimised for datasets with a shape like Video 45, as these consisted of a greater portion of the training-set. A More diverse and larger dataset would reduce this.

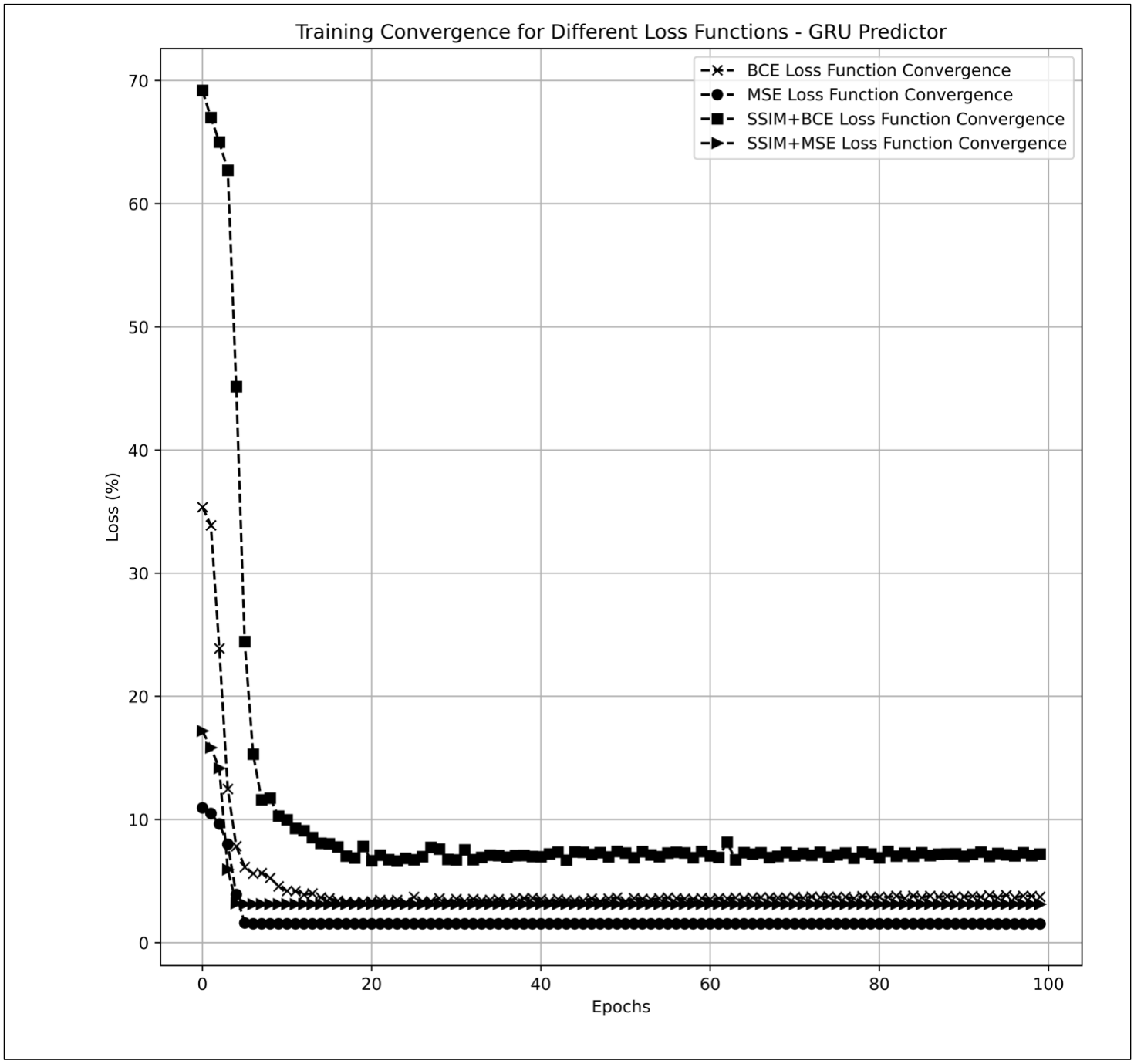
The minimal accuracy improvement compared to increase in load and ubiquity means that a standard BCE method is likely better. The limitations of this model are most definitely due to overfitting and lack of significant training diversity/size resulting in skewed model accuracy which limits the ability of the use of SSIM in improving accuracy.

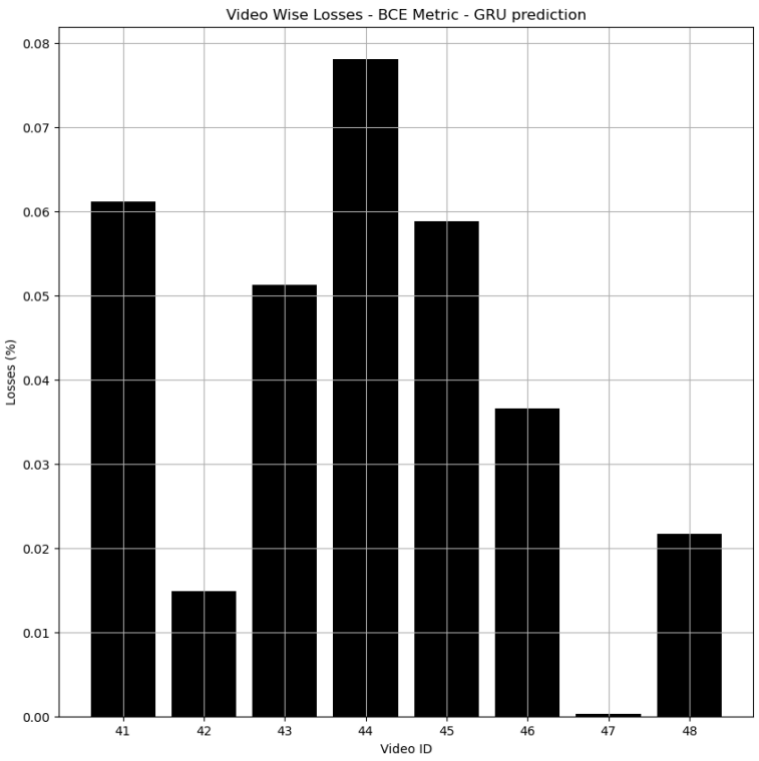
APPENDIX

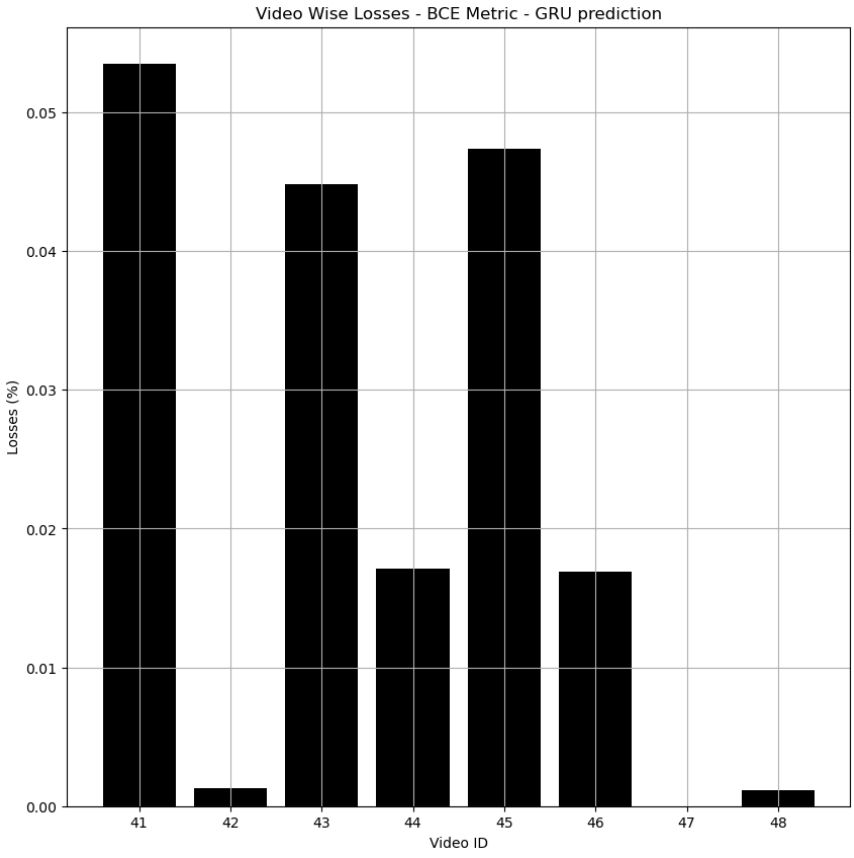

Video Wise Losses - BCE Metric - LSTM prediction
Losses (%)
0.00
0.01
0.02
0.03
0.04
0.05
0.06
0.07
41
42
43
44
45
46
47
48
Video ID
Video Wise Losses - MSE Metric - LSTM prediction
Losses (%)
0.00
0.01
0.02
0.03
0.04
0.05
41
42
43
44
45
46
47
48
Video ID


Video Wise Losses - BCE+SSIM Metric - LSTM prediction
Losses (%)
0.00
0.02
0.04
0.06
0.08
0.10
41
42
43
44
45
46
47
48
Video ID
Video Wise Losses - MSE+SSIM Metric - LSTM prediction
Losses (%)
0.00
0.01
0.02
0.03
0.04
0.05
41
42
43
44
45
46
47
48
Video ID

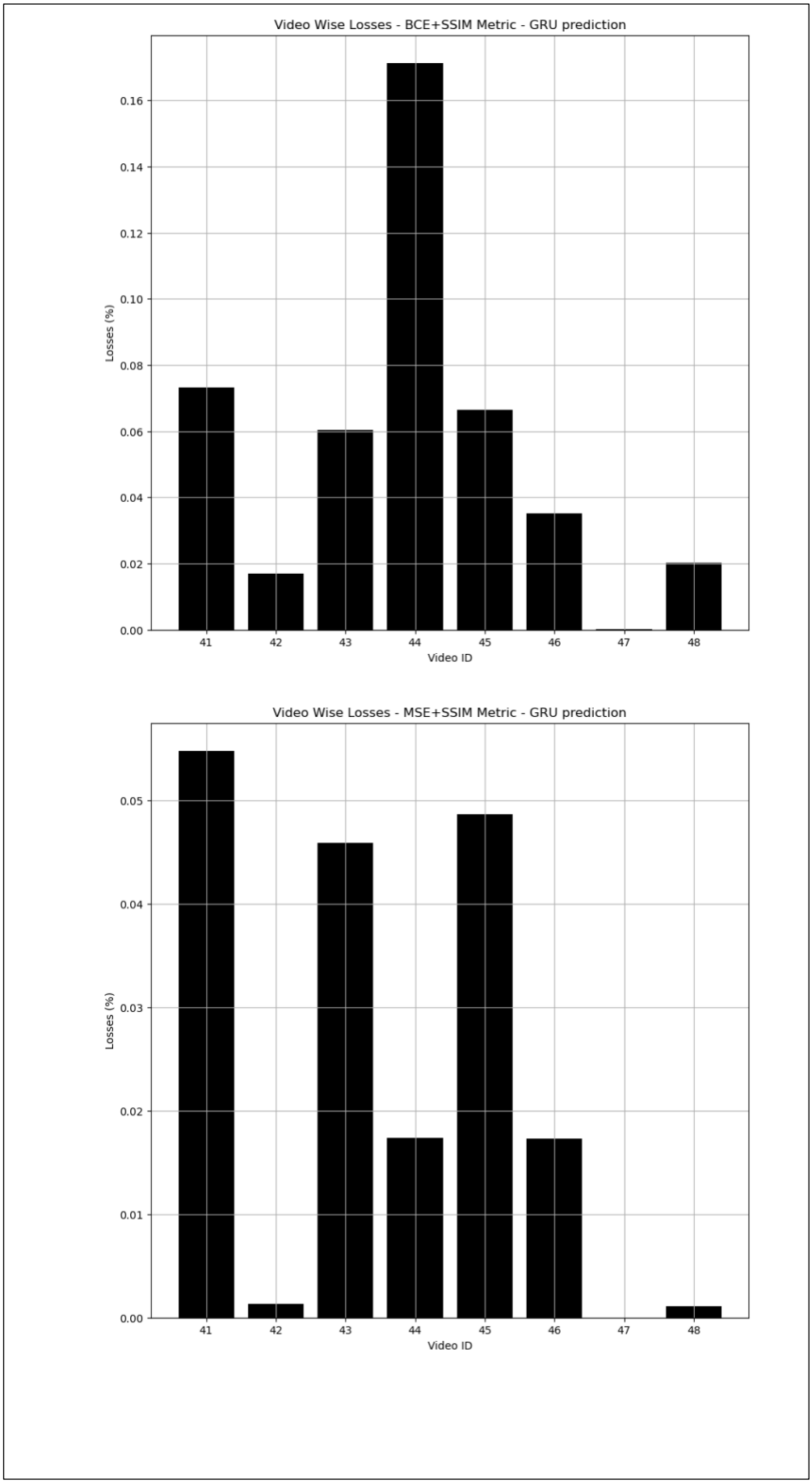

Training Convergence for Different Loss Functions - LSTM Predictor
BCE Loss Function Convergence
MSE Loss Function Convergence
SSIM+BCE Loss Function Convergence
SSIM+MSE Loss Function Convergence
Loss (%)
Epochs
0
10
20
30
40
50
60
70
0
20
40
60
80
100

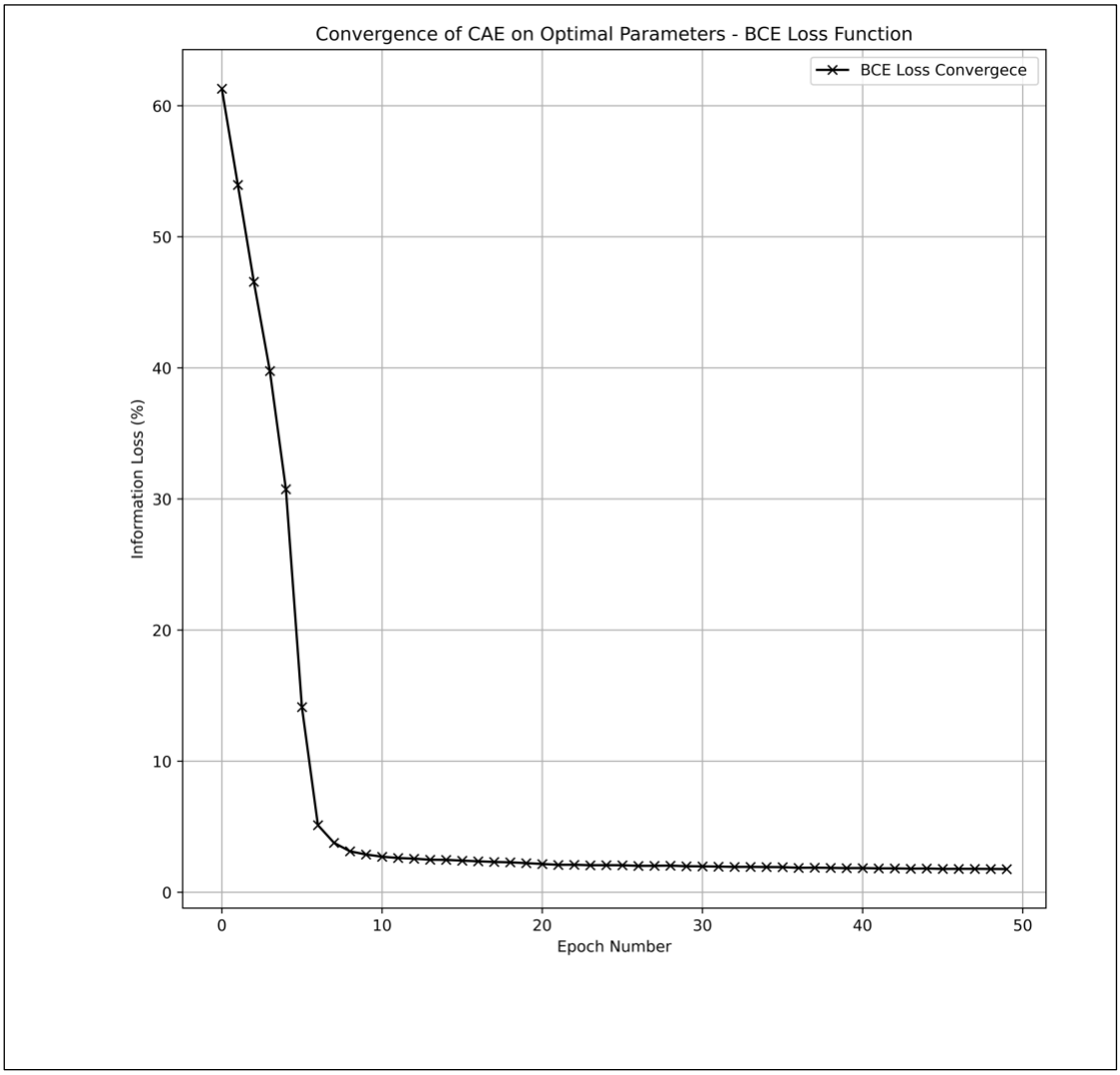

Training Convergence for Different Loss Functions - GRU Predictor
BCE Loss Function Convergence
MSE Loss Function Convergence
SSIM+BCE Loss Function Convergence
SSIM+MSE Loss Function Convergence
Loss (%)
Epochs
0
10
20
30
40
50
60
70
0
20
40
60
80
100





The above is MSE Metric, not BCE.


Video Wise Losses - BCE+SSIM Metric - GRU prediction
Losses (%)
0.00
0.02
0.04
0.06
0.08
0.10
0.12
0.14
0.16
41
42
43
44
45
46
47
48
Video ID
Video Wise Losses - MSE+SSIM Metric - GRU prediction
Losses (%)
0.00
0.01
0.02
0.03
0.04
0.05
41
42
43
44
45
46
47
48
Video ID


Convergence of CAE on Optimal Parameters - BCE Loss Function
BCE Loss Convergece
Information Loss (%)
Epoch Number
0
10
20
30
40
50
60