How Well Do the Models Do Their Jobs? - Edison

Edison - CNN

Although both versions (3 and 5 genre) of the CNN perform above random guessing, I would be hesitant to put them in any sort of automated system for the categorization of game genres. I would be more confident if the models were less confident about their choices, but they end up having high outputs for the incorrect genres quite often. It should also be noted that in the diagrams below, my (Edison) visualizations are using a the weights from the iteration of the train loop with the lowest loss.

Edison - RNN

With the LSTM added into the CNN's framework, results are a lot better, and could potentially be improved further with a longer sequence length. I could see it being put into use for an automated classification system, maybe with minor human intervention, especially for the 3 genre model. The reason I think it still needs some level of human intervention is because of the anomaly with its very high confidence that Puyo Puyo Tetris is not a puzzle game, while being very confident that Tetris effect is a puzzle game.

Angie - CNN

This model was validated in a different way to Edison's models, but it still is able to answer the question, albiet in a different manner. Among games within its training set, it was able to perform very well, showing that it could at least identify games quite well. For example, this approach could work with a screenshot aggregation website. If a user bulk uploads many screenshots, they wouldn't want to manually go through a large list, so this model could help with that. In addition to that, something like a screenshot aggregation website would have a good variety of images, so it should especially perform well there.

Overall

I would Edison's RNN and Angie's CNN ready for limited production use, though very much not on anything mission critical, due to their less than perfect accuracy when not viewing games in their dataset. Apart from the already mentioned application, I could see similar models being used for category suggestions for gameplay sharing websites.

Visualizations - Angie

Training vs. Testing Loss (Before Augmentation)

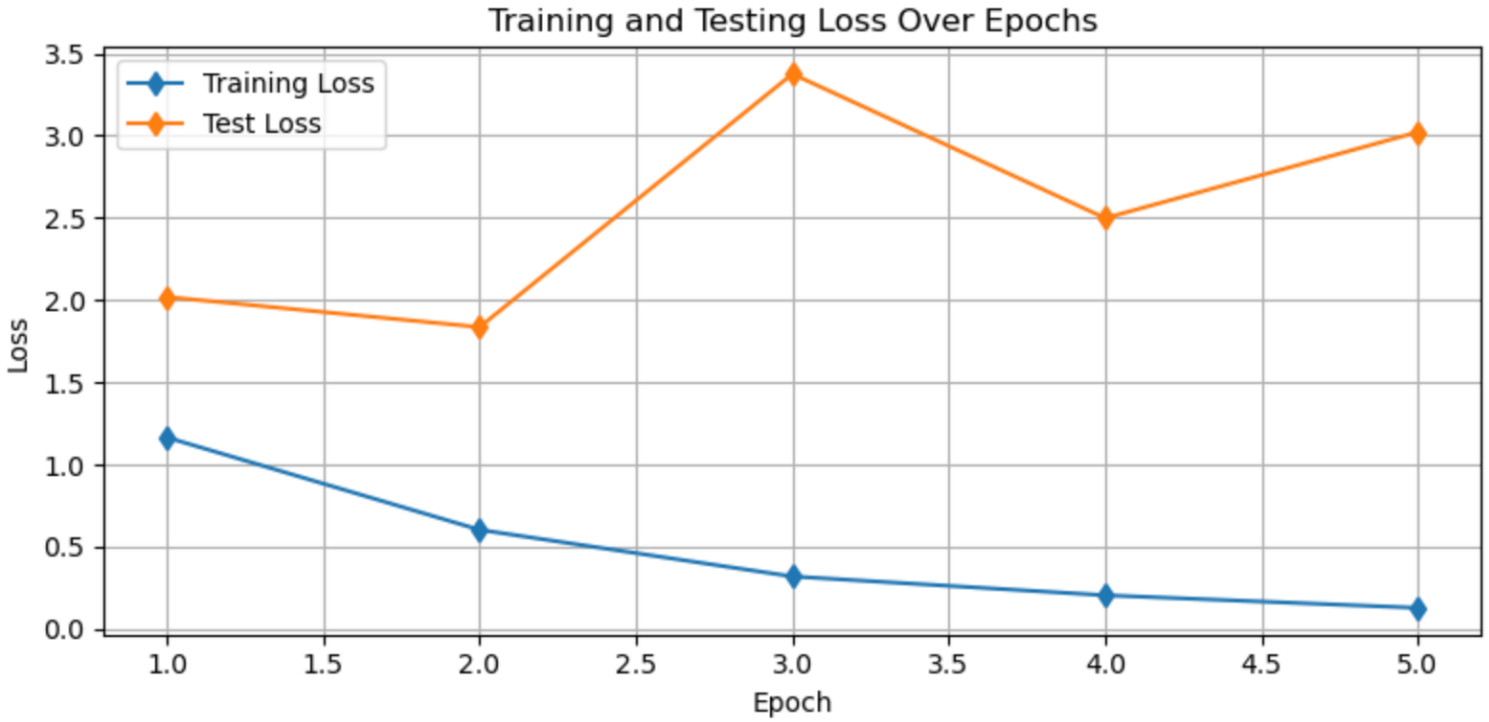

Figure 1. Training and testing loss over 5 epochs before applying data augmentation.

Before augmentation, the training loss drops quickly while the test loss remains high and unstable, including a sharp spike at epoch 3. This behavior indicates overfitting, where the model memorizes the training data rather than learning generalizable features.

Training vs. Testing Loss (After Augmentation)

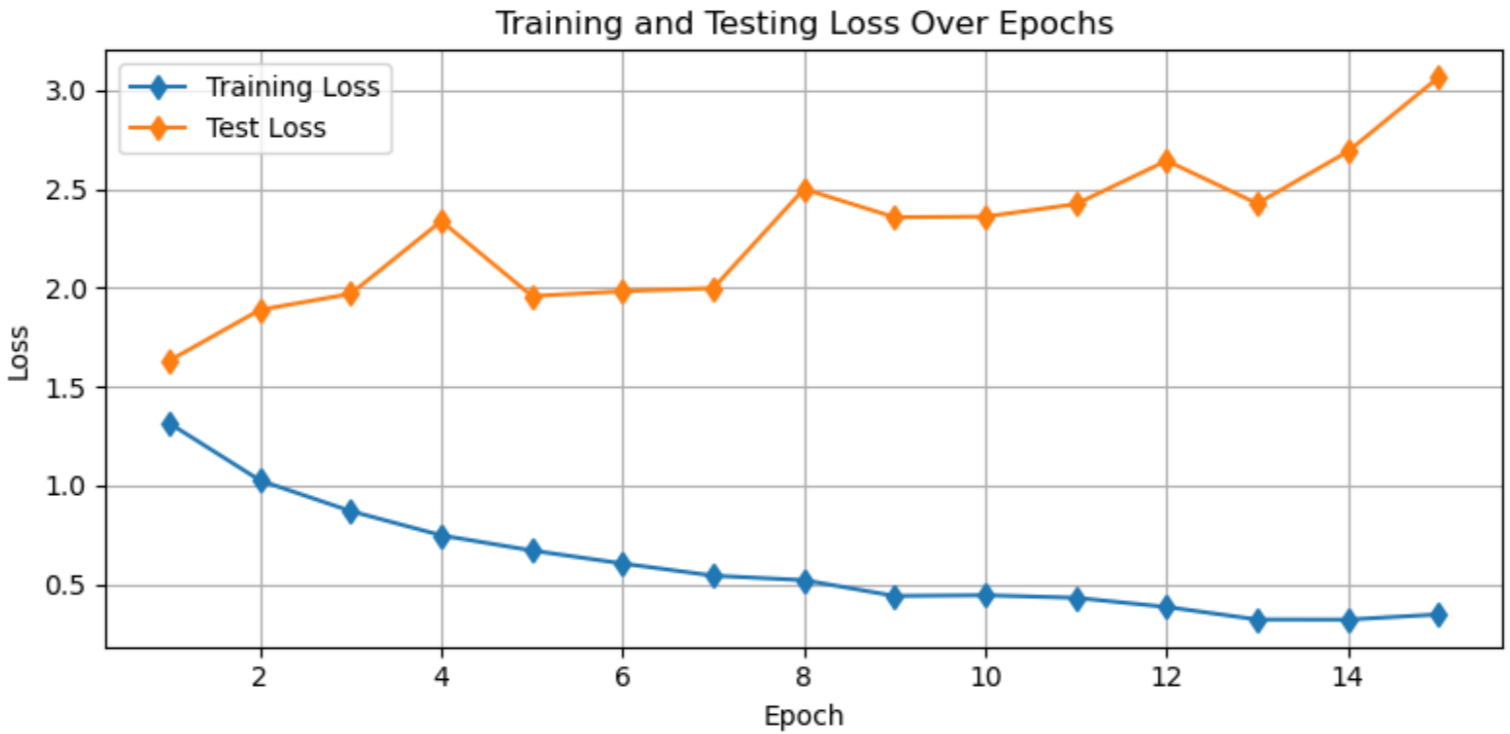

Figure 2. Training and testing loss over 15 epochs after applying data augmentation.

After augmentation, the training loss still steadily decreases, but the test loss becomes more stable and shows fewer severe spikes. Although the test loss remains relatively high, the model is forced to generalize better because the augmented images add variability. This shows reduced overfitting compared to the first model, even though perfect generalization is not reached.

Predicted Labels for Validation Images

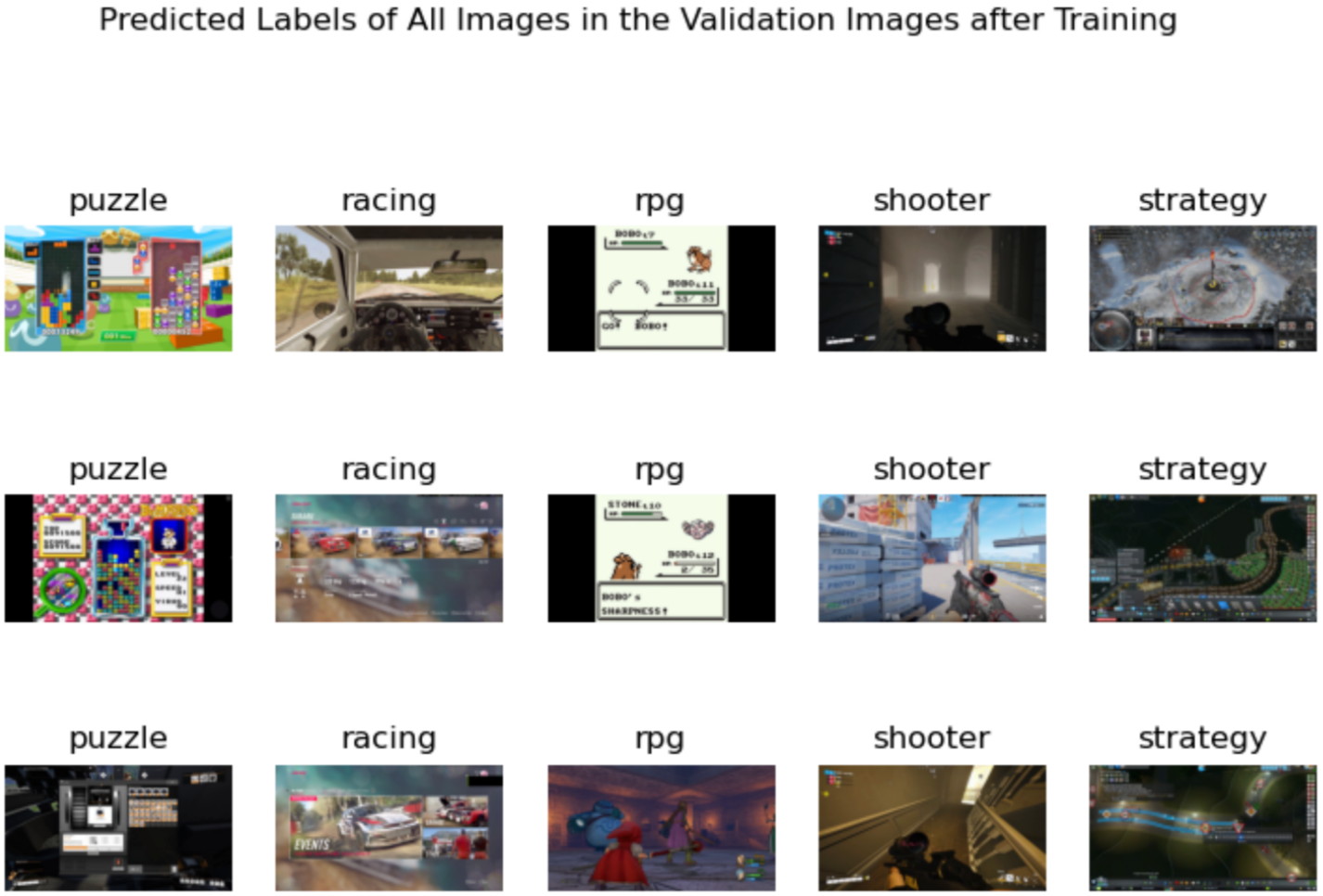

Figure 3. Model predictions for images in the validation set. Each column corresponds to a different genre, and each image is labeled with the predicted class.

These predictions demonstrate how the trained model interprets visual features from new gameplay screenshots. Genres with distinctive visuals (such as puzzle games' bright colors or strategy games' overhead views) are classified more accurately. Errors typically occur in visually similar categories (e.g., RPG vs. shooter), highlighting the challenges of learning genre-specific visual patterns.

Confusion Matrix for Genre Classification

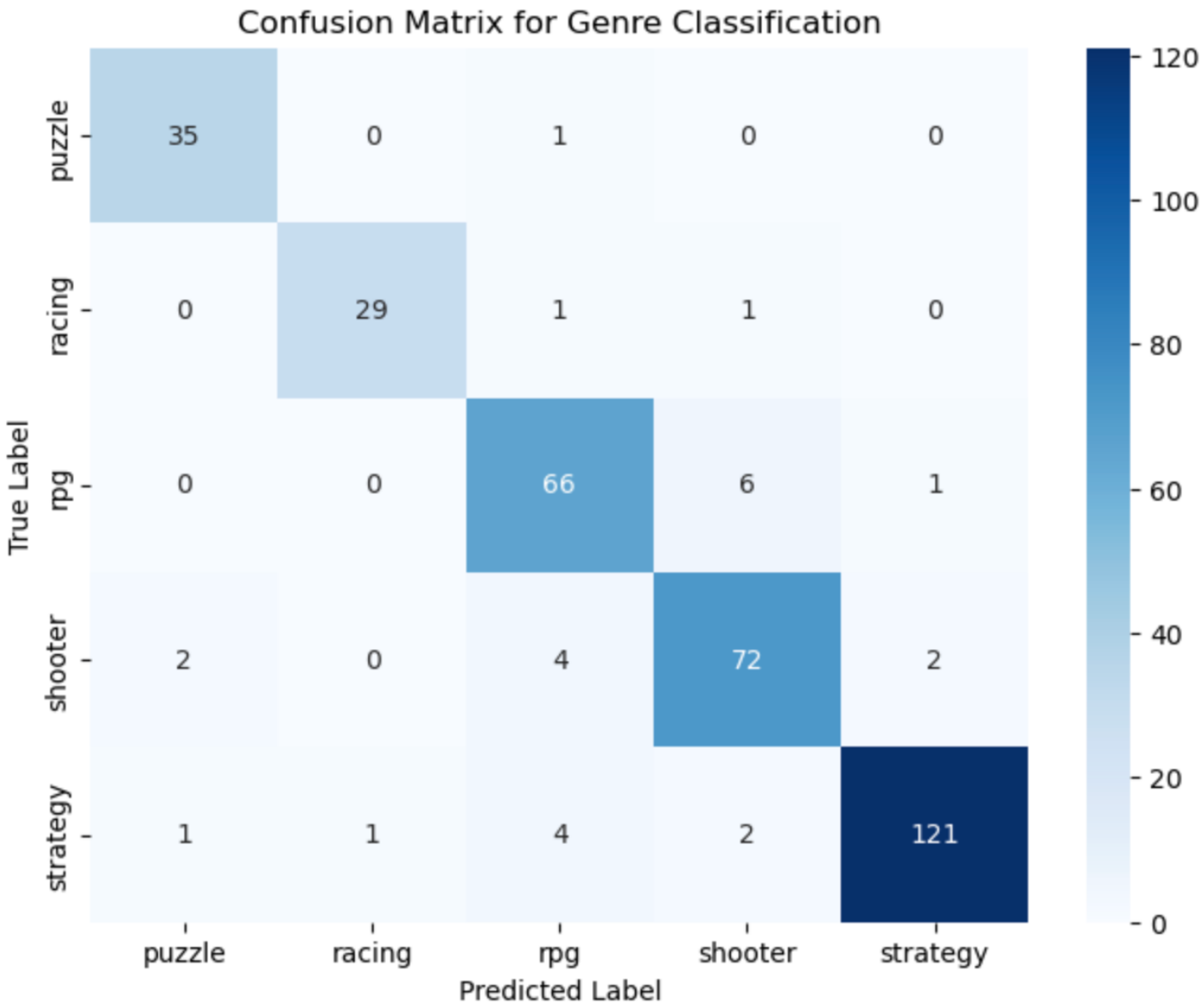

Figure 4. Confusion matrix showing the number of validation images predicted for each genre. Darker colors indicate higher

counts along the diagonal, representing correct predictions.

The confusion matrix shows that the model performs very well on puzzle, racing, and strategy images, with strong diagonal values indicating accurate classification. However, it struggles more with RPG and shooter images, which are sometimes confused with each other. This is likely due to similar camera perspectives or action-heavy scenes. Overall, the matrix highlights which genres are easiest to recognize and where the model's weaknesses lie.

RNN Prediction Grid (Pure Action, RPG, Strategy)

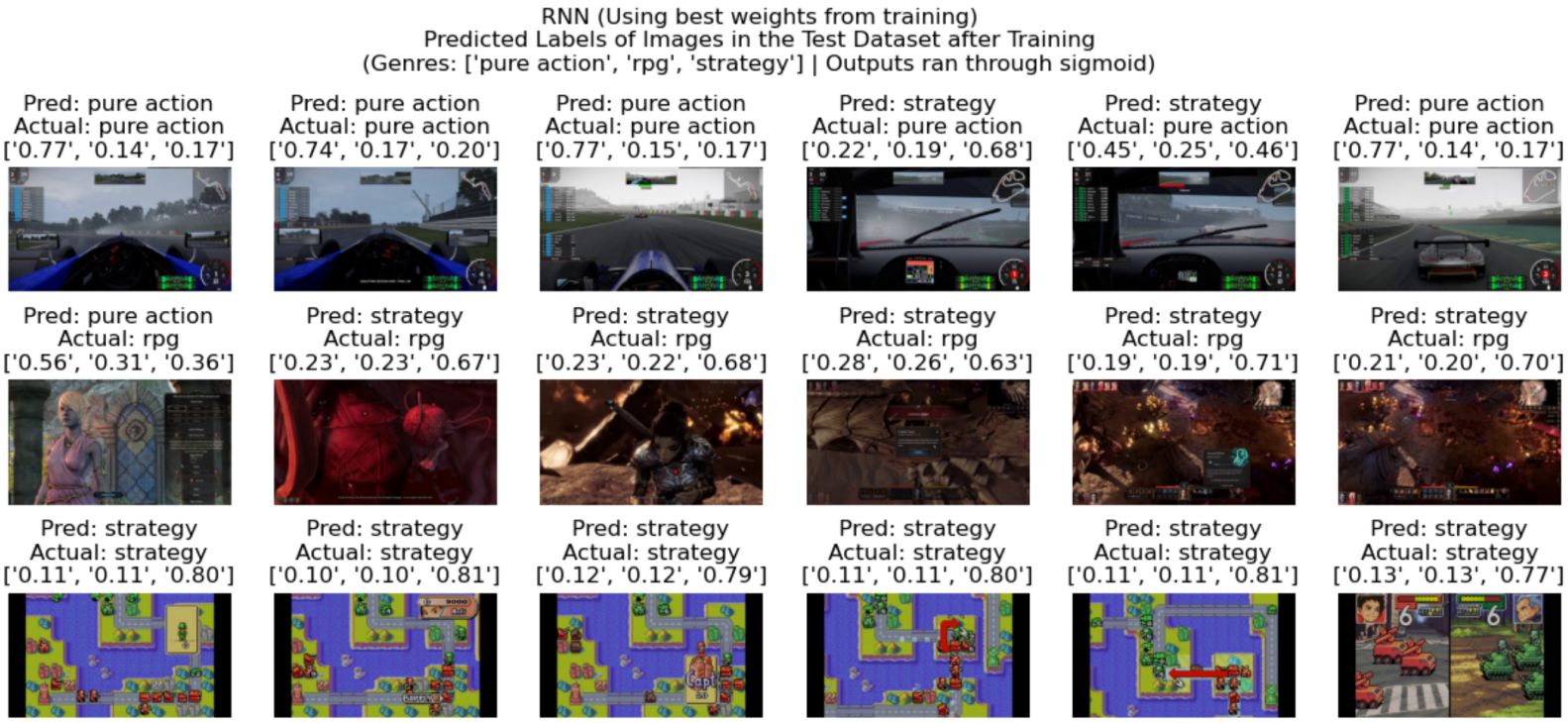

Figure 5. Grid of RNN predictions on test images using three genres. Each tile shows the predicted label, true label, and the model's output probabilities.

This figure shows the RNN's predicted labels for test images after training on a subset of three genres. While the model correctly identifies several pure action it struggles significantly when distinguishing RPG from strategy.

RNN Loss Curve (pure action, rpg, strategy)

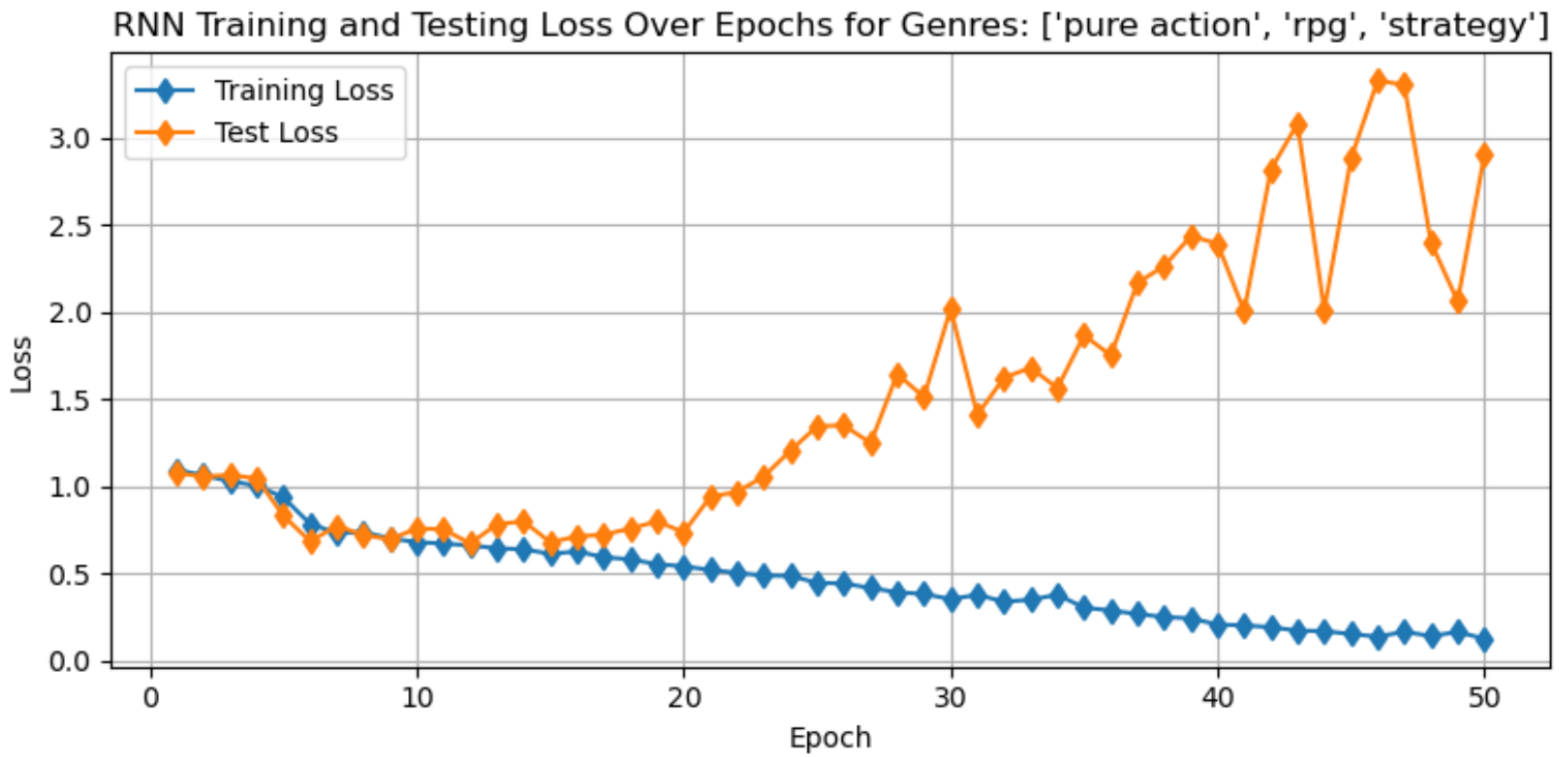

Figure 6. Training and testing loss curves for the RNN over 50 epochs. The training loss continually decreases while test loss begins rising, indicating overfitting.

This plot shows the RNN's training and testing loss over 50 epochs for a subset of genres. The training curve steadily decreases, indicating the model is learning patterns from the training data. In contrast, the test loss begins increasing after ~15 epochs, revealing clear overfitting.

RNN Training and Testing Loss (Genres: shooter, puzzle, rpg, strategy, racing)

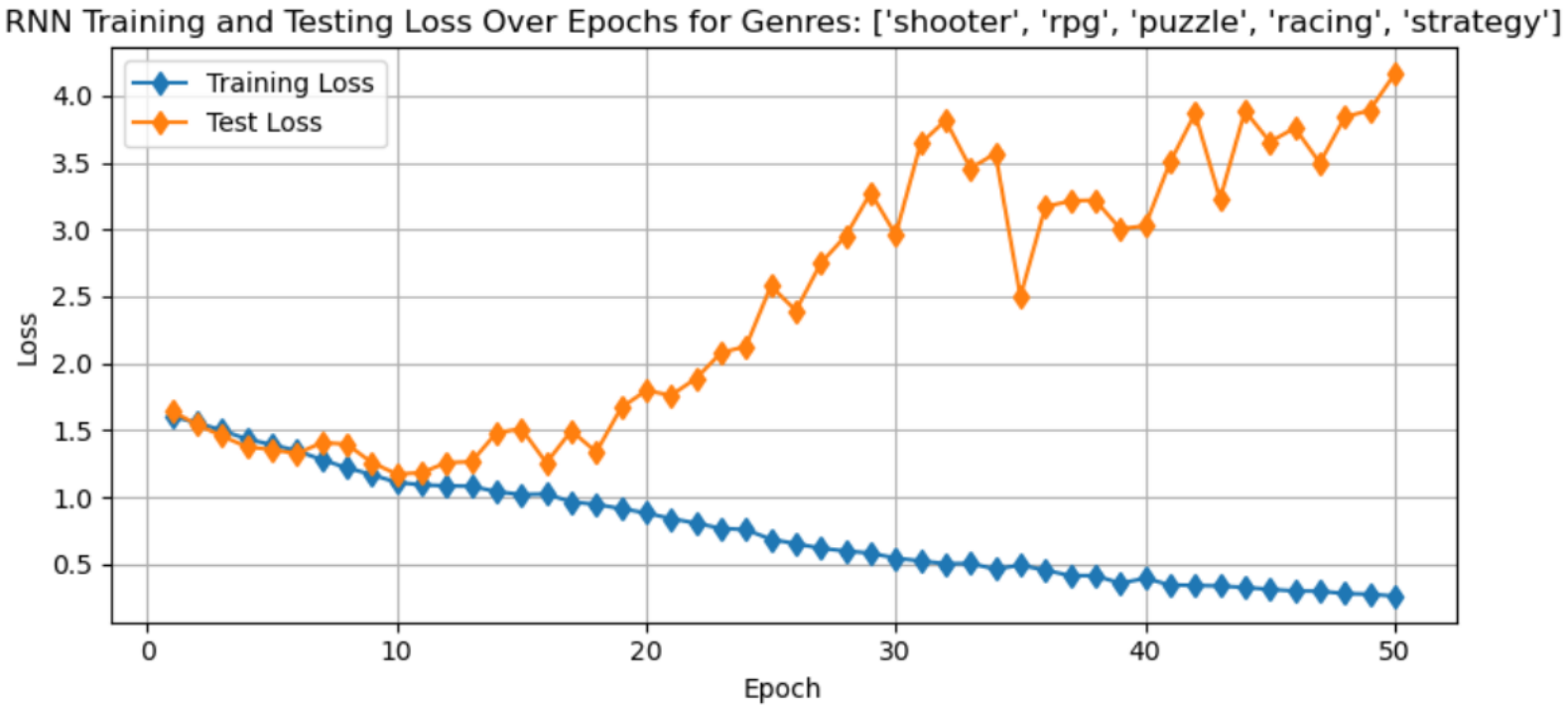

Figure 7. RNN training and testing loss curves for five genres. Test loss becomes unstable and rises early, indicating severe overfitting.

When trained on the full genre set, the RNN shows even stronger overfitting. Training loss decreases smoothly, but test loss rises sharply as training progresses. This indicates the RNN struggles to model the visual structure of screenshot data.

RNN Predicted Labels for Test Images (Genres: shooter, rpg, puzzle, racing, strategy)

RNN (Using best weights from training)
Predicted Labels of Images in the Test Dataset after Training
(Genres: ['shooter', 'rpg', 'puzzle', 'racing', 'strategy'] | Outputs ran through sigmoid)

Pred: shooter
Actual: shooter
['0.75', '0.43', '0.35', '0.72', '0.25']

Pred: strategy
Actual: shooter
['0.63', '0.54', '0.15', '0.10', '0.77']

Pred: shooter
Actual: shooter
['0.91', '0.85', '0.07', '0.51', '0.16']

Pred: shooter
Actual: shooter
['0.70', '0.44', '0.28', '0.54', '0.44']

Pred: strategy
Actual: rpg
['0.62', '0.54', '0.16', '0.11', '0.76']

Pred: strategy
Actual: rpg
['0.61', '0.53', '0.17', '0.12', '0.75']

Pred: strategy
Actual: rpg
['0.62', '0.53', '0.16', '0.12', '0.76']

Pred: strategy
Actual: rpg
['0.62', '0.53', '0.16', '0.12', '0.76']

Pred: rpg
Actual: puzzle
['0.90', '0.98', '0.09', '0.31', '0.02']

Pred: rpg
Actual: puzzle
['0.88', '0.98', '0.12', '0.30', '0.02']

Pred: rpg
Actual: puzzle
['0.90', '0.98', '0.08', '0.31', '0.03']

Pred: puzzle
Actual: puzzle
['0.56', '0.49', '0.94', '0.71', '0.01']

Pred: shooter
Actual: racing
['0.73', '0.42', '0.33', '0.64', '0.33']

Pred: shooter
Actual: racing
['0.73', '0.41', '0.34', '0.65', '0.31']

Pred: shooter
Actual: racing
['0.71', '0.43', '0.30', '0.56', '0.40']

Pred: racing
Actual: racing
['0.80', '0.35', '0.57', '0.81', '0.08']

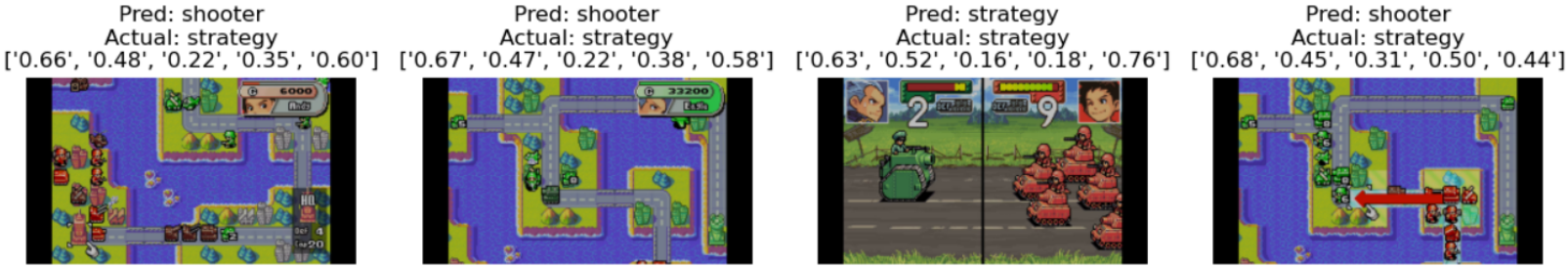

Figure 8. Grid of RNN predictions on the five genres. Predictions include confidence probabilities and true labels for each test image.

The model shows several correct classifications for shooterimages, but the model frequently misclassifies genres such as RPG and strategy. Many predictions show incorrect labels. With five genres, the RNN struggles even more.

CNN Predicted Labels for Test Images (Genres: pure action, strategy, rpg)

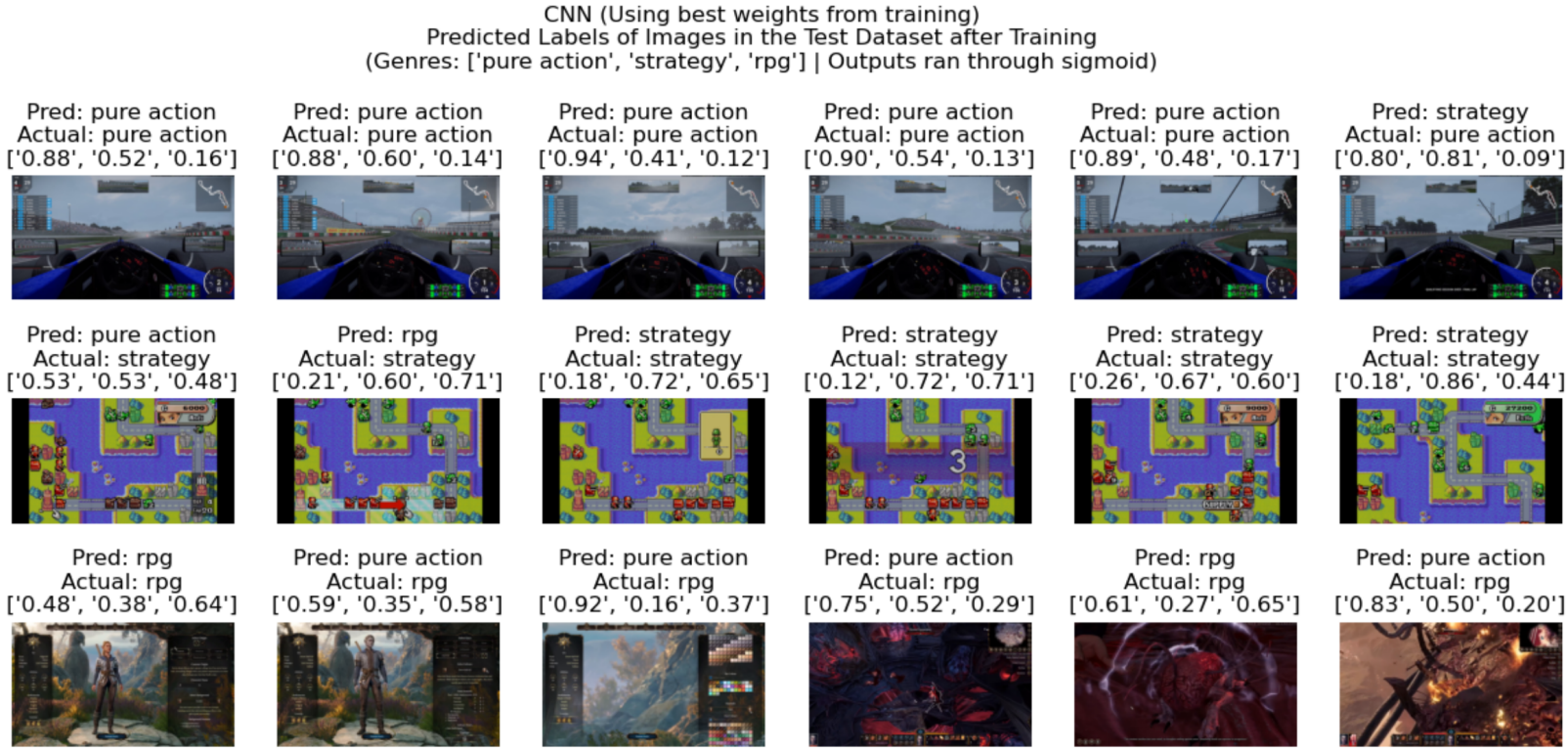

Figure 9. Grid of CNN predictions for three genres. Each image shows the predicted and true labels along with output probabilities.

Compared to the RNN, the CNN makes fewer miclassifications. The model correctly identifies most pure action and strategy images.

CNN Training and Testing Loss (Genres: pure action, strategy, rpg)

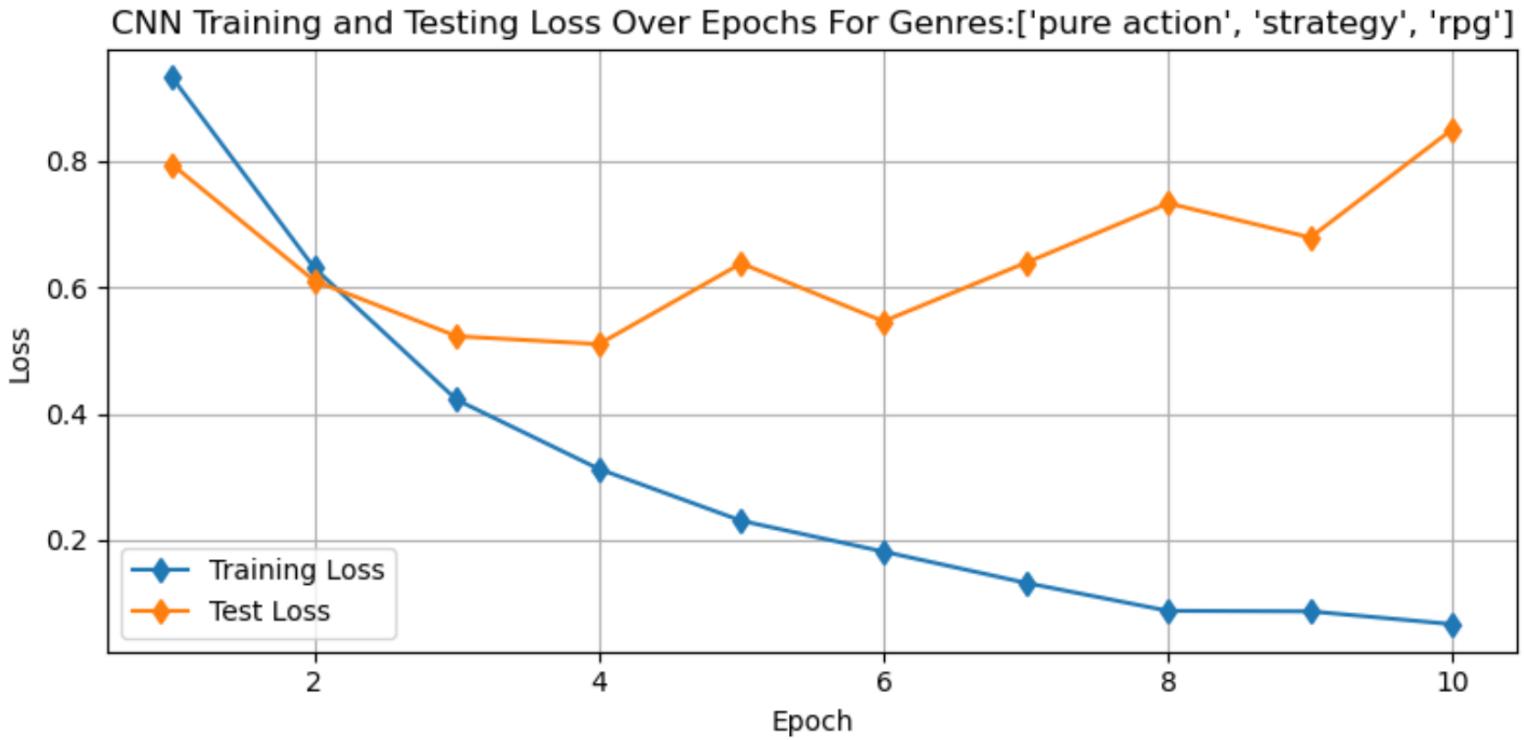

Figure 10. CNN loss curves showing stable test loss and steadily decreasing training loss. The smaller gap between curves suggests good generalization.

Compared to the RNN, the CNN models training loss decreases steadily while test loss increases only mildly. Some overfitting is present, but its magnitude is smaller

CNN Training and Testing Loss (Genres: pure action, strategy, rpg)

CNN (Using best weights from training)
Predicted Labels of Images in the Test Dataset after Training
(Genres: ['shooter', 'puzzle', 'rpg', 'strategy', 'racing'] | Outputs ran through sigmoid)

Pred: shooter
Actual: shooter
['0.72', '0.48', '0.36', '0.42', '0.57']

Pred: shooter
Actual: shooter
['0.67', '0.54', '0.34', '0.41', '0.61']

Pred: puzzle
Actual: shooter
['0.61', '0.64', '0.58', '0.35', '0.37']

Pred: shooter
Actual: shooter
['0.63', '0.41', '0.52', '0.60', '0.35']

Pred: racing
Actual: puzzle
['0.79', '0.55', '0.06', '0.61', '0.81']

Pred: puzzle
Actual: puzzle
['0.74', '0.88', '0.06', '0.34', '0.77']

Pred: puzzle
Actual: puzzle
['0.59', '0.85', '0.09', '0.58', '0.59']

Pred: puzzle
Actual: puzzle
['0.57', '0.87', '0.09', '0.62', '0.55']

Pred: shooter
Actual: rpg
['0.81', '0.42', '0.43', '0.33', '0.50']

Pred: shooter
Actual: rpg
['0.74', '0.27', '0.65', '0.69', '0.20']

Pred: shooter
Actual: rpg
['0.80', '0.23', '0.61', '0.59', '0.28']

Pred: shooter
Actual: rpg
['0.84', '0.23', '0.78', '0.33', '0.27']

Pred: puzzle
Actual: strategy
['0.39', '0.87', '0.55', '0.37', '0.30']

Pred: puzzle
Actual: strategy
['0.69', '0.81', '0.60', '0.18', '0.28']

Pred: puzzle
Actual: strategy
['0.47', '0.85', '0.58', '0.30', '0.30']

Pred: puzzle
Actual: strategy
['0.44', '0.89', '0.57', '0.34', '0.21']

Figure 11. CNN predictions on the five-genre dataset

The CNN handles the five-genre task far better than the RNN. Puzzle and shooter images are identified more often, while rpg, racing, and strategy still show confusion.

CNN Training and Testing Loss (Genres: shooter, puzzle, rpg, strategy, racing)

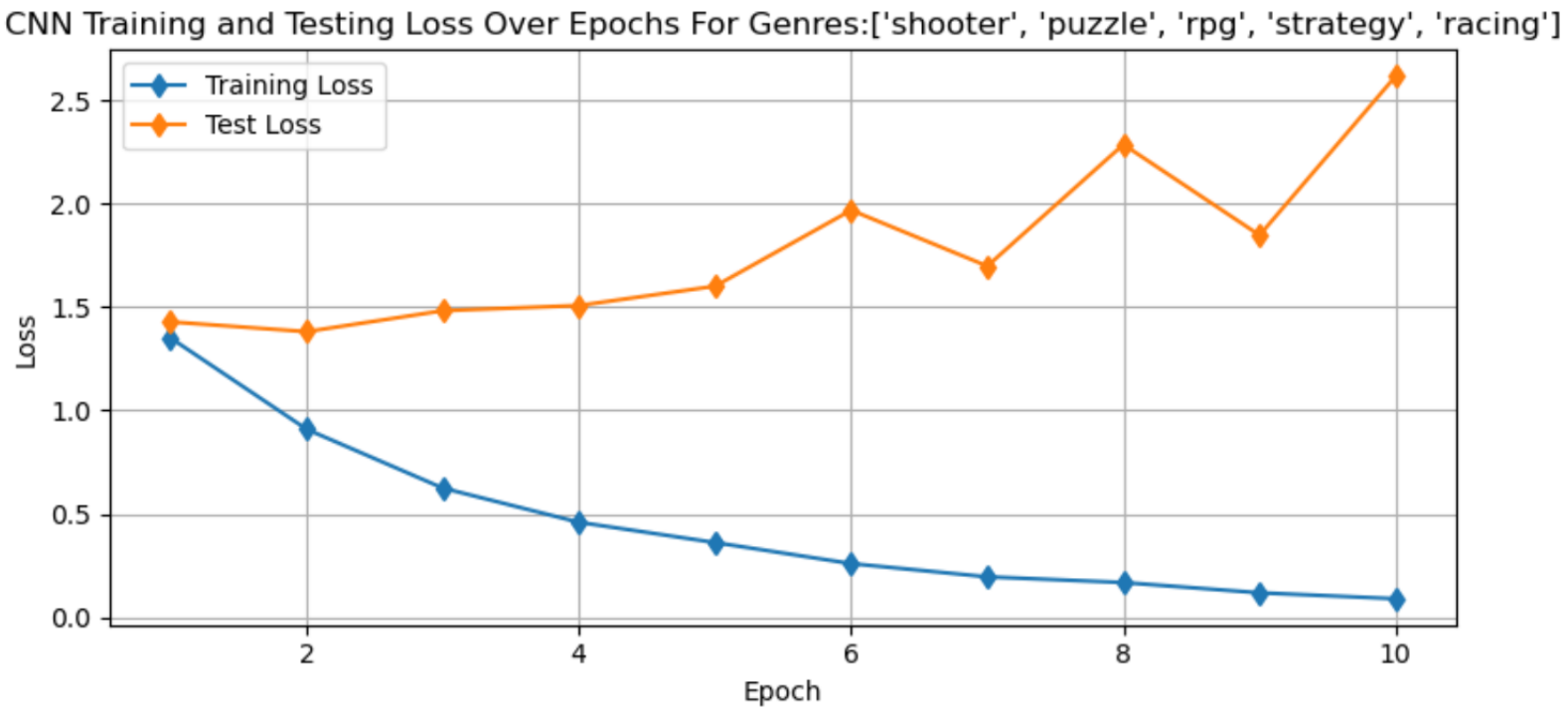

Figure 12. CNN training and testing loss curves for the five genres.

This plot displays the CNN's training and testing loss across all five genres. Training loss decreases smoothly, while test loss fluctuates and rises after several epochs, indicating overfitting. However, the gap between curves is significantly smaller than the RNN's.