IPL2023 PLAYERS IMAGE CLASSIFICATION

Introduction:

The Indian Premier League (IPL) 2023 Players Images Dataset is a comprehensive collection of images featuring players participating in the prestigious IPL cricket tournament. With cricket being one of the most beloved sports in India and around the world, the IPL has garnered immense popularity, attracting top-notch players from various countries. This dataset provides a valuable resource for researchers, data enthusiasts, and cricket fans alike to explore and analyze images of IPL players.

Dataset Description:

Key features of the dataset include:

Images: The dataset contains a diverse range of images showcasing players from different teams participating in the IPL. These images may include action shots, player portraits, team celebrations, and more.

Team Information: The dataset covers players from all participating teams in the IPL, offering a comprehensive view of the tournament's player roster. This includes players from renowned teams such as Chennai Super Kings, Mumbai Indians, Royal Challengers Bangalore, and others.

Size and Format: The dataset may vary in size and format, with images typically stored in common formats such as JPEG or PNG. The total number of images and accompanying metadata may vary depending on the version and source of the dataset.

Model Selection and Architecture:

The ResNet-18 model is loaded using models.resnet18(pretrained=True). ResNet-18 is a convolutional neural network architecture known for its effectiveness in image recognition tasks.
By setting pretrained=True, the weights of the model are initialized with pre-trained values learned on the ImageNet dataset.

Model Training:

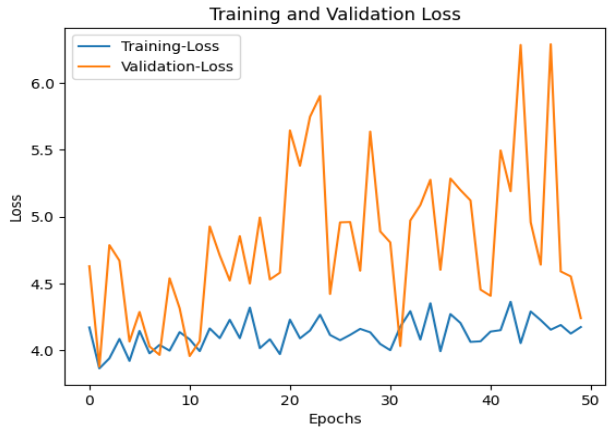
For this task, I chose CrossEntropyLoss since it works well when dealing with multiple classes. This function helps us to understand the difference between what our model predicts and what the actual classes are. To make my model learn better, I use an optimizer called Adam. It's well-liked because it adjusts how fast it learns based on how the training is going, making it efficient. I set its learning rate, which controls how much my model learns from each training step, to 0.02.

The training process iterates over the dataset for 50 epochs, updating the model's parameters using computed losses and optimizer gradients. Loss and accuracy metrics are tracked for visualization, and the model's performance is evaluated on the validation dataset. Confusion matrices are generated to assess performance visually, and the process concludes upon completing all epochs.

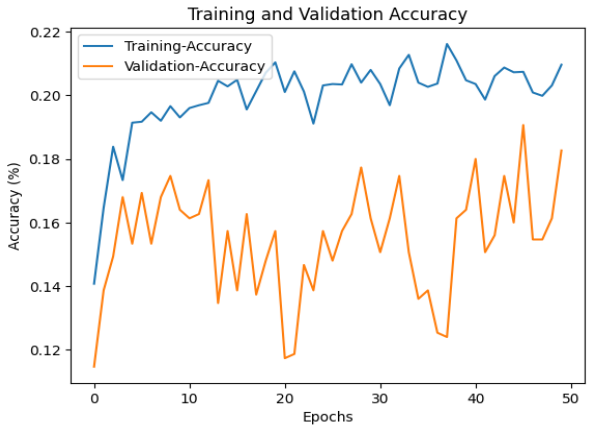
Evaluation Results:

After 50 epochs of training on the dataset containing 10 classes, the model achieved a training accuracy of 18.27% and a validation accuracy of 20.96%. The corresponding training and validation losses were 4.24 and 4.17, respectively. Additionally, the F1 score of the model was 0.16. These evaluation metrics provide insights into the model's performance in classification tasks after 50 epochs of training, indicating the need for further optimization to improve accuracy and reduce loss.

Training and Validation Loss



Training and Validation Accuracy



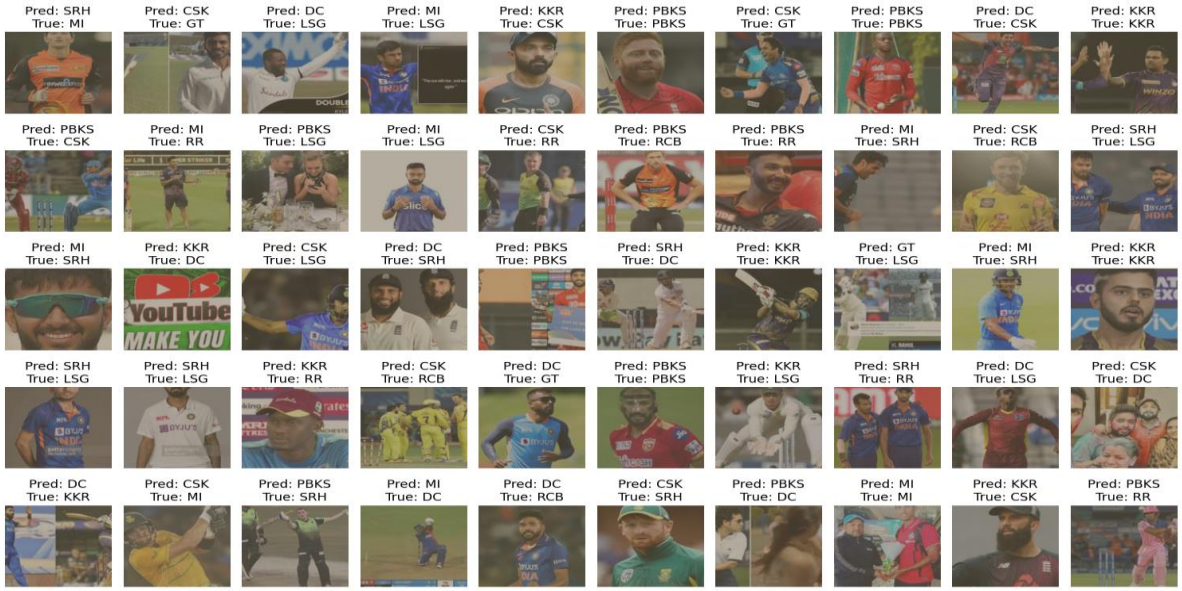
Classification Report:

```
Test Accuracy: 0.1813
--------------- Classification Report -----------------
              precision    recall  f1-score   support

         CSK       0.22      0.26      0.23        82
          DC       0.17      0.23      0.19        79
          GT       0.25      0.09      0.13        69
         KKR       0.21      0.18      0.19        83
         LSG       0.25      0.05      0.08        81
          MI       0.16      0.36      0.22        76
        PBKS       0.19      0.38      0.25        74
         RCB       0.00      0.00      0.00        66
          RR       0.44      0.06      0.10        69
         SRH       0.13      0.18      0.15        71

    accuracy                           0.18       750
   macro avg       0.20      0.18      0.16       750
weighted avg       0.20      0.18      0.16       750
```

The model's performance on the test dataset varies across classes, with notable differences in precision, recall, and F1-score. While the "RR" class demonstrates the highest precision at 0.44, its recall and F1-score are relatively lower at 0.06 and 0.10, respectively. Conversely, the "RCB" class shows the lowest performance, with all metrics at 0.00, indicating challenges in accurate prediction. Overall accuracy stands at 18%, suggesting room for improvement in classification capabilities. Both macro and weighted average F1-scores, at 0.16, highlight the need for further refinement, especially in addressing precision and recall disparities across classes.

IPL2023 PLAYERS IMAGE CLASSIFICATION

Final Tested Images:



Confusion Matrix:

The confusion matrix visually summarizes the model's classification performance across different classes. Each cell indicates the frequency of instances where the true label (y-axis) and predicted label (x-axis) intersect. Darker shades represent higher frequencies, with the diagonal showing correctly classified instances and off-diagonal elements indicating misclassifications. This matrix helps identify the model's strengths and weaknesses in predicting each class label.

Conclusion:

To conclude, although the initial training phase showed some progress, there's still a need for improvement in accuracy and reduction in loss. The evaluation on the test dataset highlighted differences among classes and underscored the necessity for refining the model's classification abilities. Further optimization is crucial to boost the overall performance and effectiveness of the model in classification tasks.