I've checked your model compilation and callbacks, and they seem to be generally well-configured. Here's a breakdown with some minor suggestions:

Callbacks:

- **ModelCheckpoint**: Good. You're saving the best model based on validation accuracy.
- **EarlyStopping**: Good. You're using early stopping to prevent overfitting and save time.
- **ReduceLROnPlateau**: Good. You're reducing the learning rate when validation accuracy plateaus, which can help the model converge better.
- **CSVLogger**: Good, you are logging the results.

Model Compilation:

- **loss**: You're using 'categorical_crossentropy', which is correct for multi-class classification with one-hot encoded labels (which you have with label_mode='categorical' in your data loading).
- **loss_weights**: You've set loss_weights=0.3. This is unusual for a single loss function. loss_weights is typically used when you have *multiple* loss functions (e.g., from multiple outputs in a multi-task learning scenario). If you only have one loss function, you should remove the loss_weights argument, or set it to 1.
- **optimizer**: You're using SGD with momentum and Nesterov acceleration, which is a good choice.
- **metrics**: You're tracking 'accuracy', which is appropriate for your classification task.

Suggestions:

- **Remove loss_weights**: Unless you have a specific reason to weight the categorical cross-entropy loss by 0.3, remove the loss_weights argument. It's not doing what you want it to do in this case.
- **Learning Rate**: Double check the learning rate. It is not defined in the code you provided. Make sure it is properly defined.

Corrected Compilation:

Here's how I recommend you compile your model:

```
model.compile(loss='categorical_crossentropy',
              optimizer=tf.keras.optimizers.SGD(learning_rate=0.01, momentum=0.9,
nesterov=True), # set learning rate
              metrics=['accuracy'])
```

Reasoning for Changes:

- **loss_weights Removed**: I removed the loss_weights argument because you only have one loss function.
- **learning_rate**: Add learning rate.

Overall, your code is well-structured. Just remove the unnecessary loss_weights argument and ensure your learning rate is correctly defined.