```
C:\Users\anama\PycharmProjects\PythonProject\.venv\Scripts\python.exe C:\Users\anama\PycharmProjects\PythonProject\Deep-fake-thesis\main.py
2025-10-01 10:46:11.407819: I tensorflow/core/util/port.cc:153] oneDNN custom operations are on. You may see slightly different numerical results due to floating-point round-off errors from different computation orders. To turn them off, set the environment variable `TF_ENABLE_ONEDNN_OPTS=0`.
2025-10-01 10:46:17.544925: I tensorflow/core/util/port.cc:153] oneDNN custom operations are on. You may see slightly different numerical results due to floating-point round-off errors from different computation orders. To turn them off, set the environment variable `TF_ENABLE_ONEDNN_OPTS=0`.
Using TensorFlow 2.20.0
Config: {
  "model_name": "efficientnetb7",
  "data_dir": "C:/Users/anama/PycharmProjects/PythonProject/Frames/Celeb-DF-600",
  "epochs": 20,
  "batch_size": 32,
  "seed": 42,
  "base_trainable_at": -40,
  "warmup_epochs": 3,
  "learning_rate": 0.001,
  "fine_tune_lr": 2e-05,
  "use_class_weights": false,
  "mixed_precision": false,
  "output_dir": "C:/Users/anama/PycharmProjects/PythonProject/Frames/Celeb-DF/Efficientnetb7/Celeb600"
}
Found 56902 images belonging to 2 classes.
Found 12197 images belonging to 2 classes.
Found 12195 images belonging to 2 classes.
2025-10-01 10:46:29.684410: I tensorflow/core/platform/cpu_feature_guard.cc:210] This TensorFlow binary is optimized to use available CPU instructions in performance-critical operations.
To enable the following instructions: SSE3 SSE4.1
```

```
SSE4.2 AVX AVX2 AVX_VNNI FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
C:\Users\anama\PycharmProjects\PythonProject\.venv\Lib\site-packages\keras\src\trainers\data_adapters\py_dataset_adapter.py:121: UserWarning: Your `PyDataset` class should call `super().__init__(**kwargs)` in its constructor. `**kwargs` can include `workers`, `use_multiprocessing`, `max_queue_size`. Do not pass these arguments to `fit()`, as they will be ignored.
  self._warn_if_super_not_called()
Epoch 1/3
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 0s 31s/step - accuracy: 0.9015 - loss: 0.3237
Epoch 1: val_accuracy improved from -inf to 0.90407, saving model to C:/Users/anama/PycharmProjects/PythonProject/Frames/Celeb-DF/Efficientnetb7/Celeb600\best_warmup.keras
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 63492s 36s/step - accuracy: 0.9015 - loss: 0.3237 - val_accuracy: 0.9041 - val_loss: 0.3165 - learning_rate: 0.0010
Epoch 2/3
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 0s 39s/step - accuracy: 0.9036 - loss: 0.3128
Epoch 2: val_accuracy improved from 0.90407 to 0.90481, saving model to C:/Users/anama/PycharmProjects/PythonProject/Frames/Celeb-DF/Efficientnetb7/Celeb600\best_warmup.keras
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 87308s 49s/step - accuracy: 0.9036 - loss: 0.3128 - val_accuracy: 0.9048 - val_loss: 0.3027 - learning_rate: 0.0010
Epoch 3/3
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 0s 49s/step - accuracy: 0.9032 - loss: 0.3105
Epoch 3: val_accuracy did not improve from 0.90481
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 99992s 56s/step - accuracy: 0.9032 - loss: 0.3105 - val_accuracy: 0.9045 - val_loss: 0.2989 - learning_rate: 0.0010
Epoch 1/20
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 0s 50s/step - accuracy
```

```
: 0.8956 - loss: 0.3146
Epoch 1: val_accuracy improved from -inf to 0.91719, saving model to C:/Users/anama/PycharmProjects/PythonProject/Frames/Celeb-DF/Efficientnetb7/Celeb600\best_finetune.keras
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 102102s 57s/step - accuracy: 0.8956 - loss: 0.3146 - val_accuracy: 0.9172 - val_loss: 0.2323 - learning_rate: 2.0000e-05
Epoch 2/20
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 0s 61s/step - accuracy: 0.9188 - loss: 0.2315
Epoch 2: val_accuracy improved from 0.91719 to 0.93802, saving model to C:/Users/anama/PycharmProjects/PythonProject/Frames/Celeb-DF/Efficientnetb7/Celeb600\best_finetune.keras
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 125390s 70s/step - accuracy: 0.9188 - loss: 0.2315 - val_accuracy: 0.9380 - val_loss: 0.1736 - learning_rate: 2.0000e-05
Epoch 3/20
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 0s 32s/step - accuracy: 0.9327 - loss: 0.1913
Epoch 3: val_accuracy improved from 0.93802 to 0.94507, saving model to C:/Users/anama/PycharmProjects/PythonProject/Frames/Celeb-DF/Efficientnetb7/Celeb600\best_finetune.keras
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 75262s 42s/step - accuracy: 0.9327 - loss: 0.1913 - val_accuracy: 0.9451 - val_loss: 0.1568 - learning_rate: 2.0000e-05
Epoch 4/20
2025-10-07 20:32:31.705735: W tensorflow/core/kernels/data/prefetch_autotuner.cc:55] Prefetch autotuner tried to allocate 138240256 bytes after encountering the first element of size 138240256 bytes.This already causes the autotune ram budget to be exceeded. To stay within the ram budget, either increase the ram budget or reduce element size
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 0s 78s/step - accuracy: 0.9385 - loss: 0.1743
Epoch 4: val_accuracy did not improve from 0.94507
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 159319s 90s/step - accuracy: 0.9385 - loss: 0.1743 - val_accuracy: 0.
```

```
9425 - val_loss: 0.1643 - learning_rate: 2.0000e-05
Epoch 5/20
2025-10-09 16:47:51.947672: W tensorflow/core/kernels/data/prefetch_autotuner.cc:55] Prefetch autotuner tried to allocate 138240256 bytes after encountering the first element of size 138240256 bytes.This already causes the autotune ram budget to be exceeded. To stay within the ram budget, either increase the ram budget or reduce element size
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 0s 60s/step - accuracy: 0.9427 - loss: 0.1596
Epoch 5: val_accuracy improved from 0.94507 to 0.95433, saving model to C:/Users/anama/PycharmProjects/PythonProject/Frames/Celeb-DF/Efficientnetb7/Celeb600\best_finetune.keras
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 116459s 65s/step - accuracy: 0.9427 - loss: 0.1596 - val_accuracy: 0.9543 - val_loss: 0.1300 - learning_rate: 2.0000e-05
Epoch 6/20
2025-10-11 01:08:56.177971: W tensorflow/core/kernels/data/prefetch_autotuner.cc:55] Prefetch autotuner tried to allocate 138240256 bytes after encountering the first element of size 138240256 bytes.This already causes the autotune ram budget to be exceeded. To stay within the ram budget, either increase the ram budget or reduce element size
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 0s 39s/step - accuracy: 0.9436 - loss: 0.1539
Epoch 6: val_accuracy did not improve from 0.95433
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 88479s 50s/step - accuracy: 0.9436 - loss: 0.1539 - val_accuracy: 0.9541 - val_loss: 0.1332 - learning_rate: 2.0000e-05
Epoch 7/20
2025-10-12 01:43:30.971857: W tensorflow/core/kernels/data/prefetch_autotuner.cc:55] Prefetch autotuner tried to allocate 138240256 bytes after encountering the first element of size 138240256 bytes.This already causes the autotune ram budget to be exceeded. To stay within the ram budget, either increase the ram budget or reduce element size
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 0s 67s/step - accuracy
```

```
: 0.9502 - loss: 0.1388
Epoch 7: val_accuracy improved from 0.95433 to 0.95925, saving model to C:/Users/anama/PycharmProjects/PythonProject/Frames/Celeb-DF/Efficientnetb7/Celeb600\best_finetune.keras
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 141209s 79s/step - accuracy: 0.9502 - loss: 0.1388 - val_accuracy: 0.9593 - val_loss: 0.1160 - learning_rate: 2.0000e-05
Epoch 8/20
2025-10-13 16:57:01.132214: W tensorflow/core/kernels/data/prefetch_autotuner.cc:55] Prefetch autotuner tried to allocate 138240256 bytes after encountering the first element of size 138240256 bytes.This already causes the autotune ram budget to be exceeded. To stay within the ram budget, either increase the ram budget or reduce element size
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 0s 59s/step - accuracy: 0.9495 - loss: 0.1360
Epoch 8: val_accuracy did not improve from 0.95925
1779/1779 ━━━━━━━━━━━━━━━━━━━━ 114477s 64s/step - accuracy: 0.9495 - loss: 0.1360 - val_accuracy: 0.9564 - val_loss: 0.1173 - learning_rate: 2.0000e-05
Epoch 9/20

```