

```
+ - ✂ 📄 ▶ ▶▶ Run All Code Draft Session off (run a cell to start) Notebook
!pip install moviepy --upgrade
!pip install gdown

Requirement already satisfied: moviepy in /usr/local/lib/python3.10/dist-packages (1.0.3)
Collecting moviepy
  Downloading moviepy-2.1.1-py3-none-any.whl.metadata (6.9 kB)
Requirement already satisfied: decorator<6.0,>=4.0.2 in /usr/local/lib/python3.10/dist-packages (from moviepy) (4.4.2)
Requirement already satisfied: imageio<3.0,>=2.5 in /usr/local/lib/python3.10/dist-packages (from moviepy) (2.35.1)
Requirement already satisfied: imageio_ffmpeg>=0.2.0 in /usr/local/lib/python3.10/dist-packages (from moviepy) (0.5.1)
Requirement already satisfied: numpy>=1.25.0 in /usr/local/lib/python3.10/dist-packages (from moviepy) (1.26.4)
Requirement already satisfied: proglog<=1.0.0 in /usr/local/lib/python3.10/dist-packages (from moviepy) (0.1.10)
Collecting python-dotenv>=0.10 (from moviepy)
  Downloading python_dotenv-1.0.1-py3-none-any.whl.metadata (23 kB)
Requirement already satisfied: pillow<11.0,>=9.2.0 in /usr/local/lib/python3.10/dist-packages (from moviepy) (10.4.0)
Requirement already satisfied: setuptools in /usr/local/lib/python3.10/dist-packages (from imageio_ffmpeg>=0.2.0->moviepy) (71.0.4)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from proglog<=1.0.0->moviepy) (4.66.5)
Downloading moviepy-2.1.1-py3-none-any.whl (123 kB)
```

```
Assignment №2 Draft saved
File Edit View Run Settings Add-ons Help
+ - ✂ 📄 ▶ ▶▶ Run All Code Draft Session off (run a cell to start) Notebook
123.5/123.5 kB 6.5 MB/s eta 0:00:00
Downloading python_dotenv-1.0.1-py3-none-any.whl (19 kB)
Installing collected packages: python-dotenv, moviepy
  Attempting uninstall: moviepy
    Found existing installation: moviepy 1.0.3
    Uninstalling moviepy-1.0.3:
      Successfully uninstalled moviepy-1.0.3
  Successfully installed moviepy-2.1.1 python-dotenv-1.0.1
Requirement already satisfied: gdown in /usr/local/lib/python3.10/dist-packages (5.2.0)
Requirement already satisfied: BeautifulSoup4 in /usr/local/lib/python3.10/dist-packages (from gdown) (4.12.3)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from gdown) (3.16.1)
Requirement already satisfied: requests[socks] in /usr/local/lib/python3.10/dist-packages (from gdown) (2.32.3)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from gdown) (4.66.5)
Requirement already satisfied: soupsieve>1.2 in /usr/local/lib/python3.10/dist-packages (from BeautifulSoup4->gdown) (2.6)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests[socks]->gdown) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests[socks]->gdown) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests[socks]->gdown) (2.2.3)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests[socks]->gdown) (2024.8.30)
Requirement already satisfied: PySocks!=1.5.7,>=1.5.6 in /usr/local/lib/python3.10/dist-packages (from requests[socks]->gdown) (1.7.1)
```

```
File Edit View Run Settings Add-ons Help
+ - ✂ 📄 ▶ ▶▶ Run All Code Draft Session off (run a cell to start) Notebook
[2]:
from pathlib import Path
from typing import List, Tuple, Sequence

import numpy as np
from numpy import unravel_index
from PIL import Image, ImageDraw, ImageFont
from tqdm import tqdm, notebook

from moviepy.video.io.ImageSequenceClip import ImageSequenceClip

import math
from scipy.ndimage import gaussian_filter

import gc
import time
import random
import csv
import matplotlib.pyplot as plt
from scipy.interpolate import interp1d
from tensorflow import keras
from tensorflow.keras import layers
from tensorflow.keras import backend as K
from tensorflow.keras.optimizers import Adam
import gdown
```

```
File Edit View Run Settings Add-ons Help
+ Run All Code Draft Session off (run a cell to start)
def get_num_clips(path: Path, game: int) -> int:
    return len(list((path / f'game{game}').iterdir()))

def get_game_clip_pairs(path: Path, games: List[int]) -> List[Tuple[int, int]]:
    return [(game, c) for game in games for c in range(1, get_num_clips(path, game) + 1)]

def load_clip_data(path: Path, game: int, clip: int, downscale: bool, quiet=False) -> np.ndarray:
    if not quiet:
        suffix = 'downscaled' if downscale else ''
        print(f'loading clip data (game {game}, clip {clip}) {suffix}')
    cache_path = path / 'cache'
    cache_path.mkdir(exist_ok=True)
    resize_code = '_ds2' if downscale else ''
    cached_data_name = f'{game}_{clip}{resize_code}.npz'
    if (cache_path / cached_data_name).exists():
        clip_data = np.load(cache_path / cached_data_name)['clip_data']
    else:
        clip_path = path / f'game{game}/clip{clip}'
        n_imgs = len(list(clip_path.iterdir())) - 1
        imgs = [None] * n_imgs
        for i in notebook.tqdm(range(n_imgs)):
            img = Image.open(clip_path / f'{i:04d}.jpg')
            if downscale:
                img = img.resize((img.width // 2, img.height // 2))
```

```
File Edit View Run Settings Add-ons Help
+ Run All Code Draft Session off (run a cell to start)
    if downscale:
        img = img.resize((img.width // 2, img.height // 2))
        imgs[i] = np.array(img, dtype=np.uint8)
    clip_data = np.stack(imgs)
    cache_path.mkdir(exist_ok=True, parents=True)
    np.savez_compressed(cache_path / cached_data_name, clip_data=clip_data)
    return clip_data

def load_clip_labels(path: Path, game: int, clip: int, downscale: bool, quiet=False):
    if not quiet:
        print(f'loading clip labels (game {game}, clip {clip})')
    clip_path = path / f'game{game}/clip{clip}'
    labels = []
    with open(clip_path / 'labels.csv') as csvfile:
        lines = list(csv.reader(csvfile))
        for line in lines[1:]:
            values = np.array([-1 if i == '' else int(i) for i in line[1:]])
            if downscale:
                values[1] //= 2
                values[2] //= 2
            labels.append(values)
    return np.stack(labels)

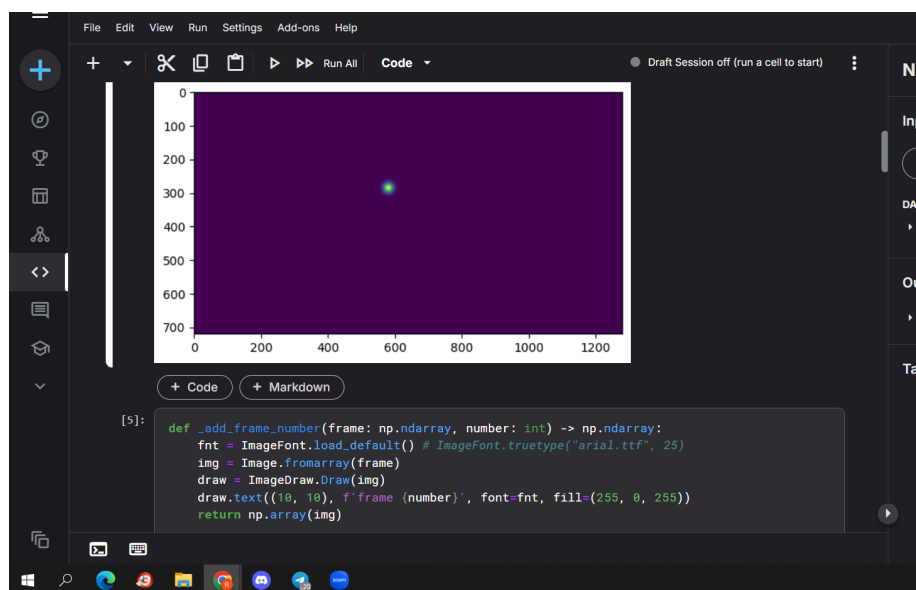
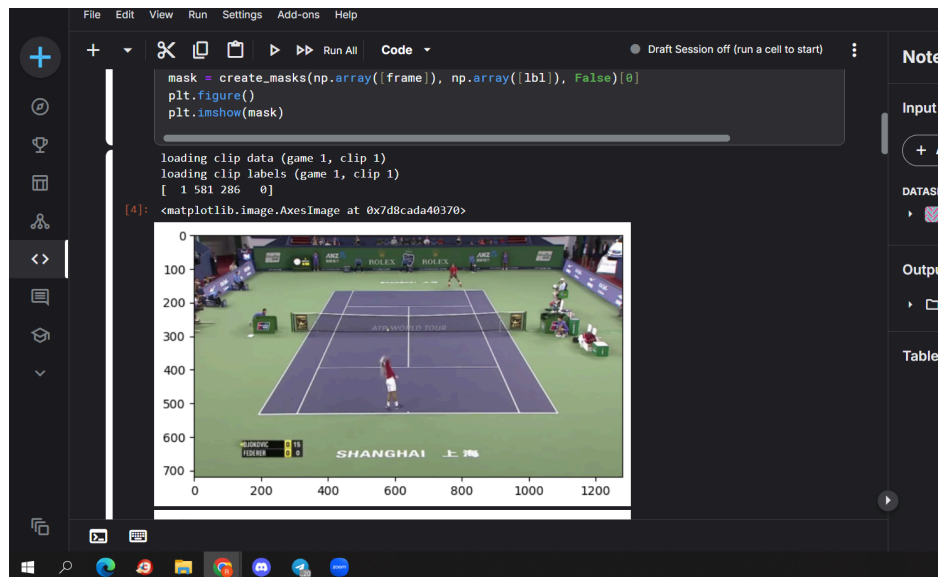
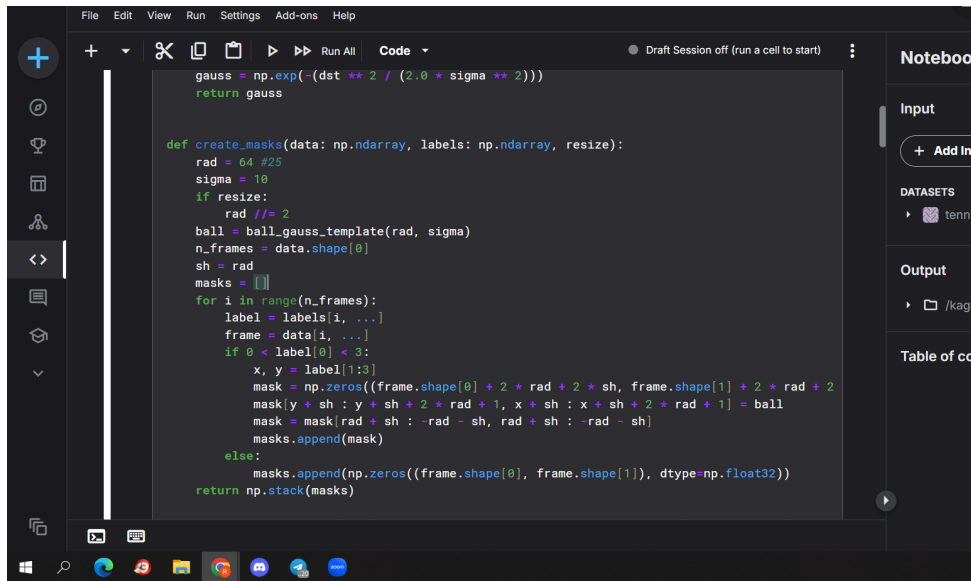
def load_clip(path: Path, game: int, clip: int, downscale: bool, quiet=False):
```

```
+ Run All Code Draft Session off (run a cell to start)
def load_clip(path: Path, game: int, clip: int, downscale: bool, quiet=False):
    data = load_clip_data(path, game, clip, downscale, quiet)
    labels = load_clip_labels(path, game, clip, downscale, quiet)
    return data, labels

def prepare_experiment(out_path: Path) -> Path:
    out_path.mkdir(parents=True, exist_ok=True)
    dirs = [d for d in out_path.iterdir() if d.is_dir() and d.name.startswith('exp_')]
    experiment_id = max(int(d.name.split('_')[1]) for d in dirs) + 1 if dirs else 1
    exp_path = out_path / f'exp_{experiment_id}'
    exp_path.mkdir()
    return exp_path

def ball_gauss_template(rad, sigma):
    x, y = np.meshgrid(np.linspace(-rad, rad, 2 * rad + 1), np.linspace(-rad, rad, 2 * rad + 1))
    dst = np.sqrt(x * x + y * y)
    gauss = np.exp(-(dst ** 2 / (2.0 * sigma ** 2)))
    return gauss

def create_masked_data: np.ndarray, labels: np.ndarray, resize:
```



```
File Edit View Run Settings Add-ons Help
+ - ✂ 📄 📁 ▶ ⏏ Run All Code
Draft Session off (run a cell to start)
Notebook
Input
+ Add Input
DATASETS
> tennist
Output
> /kaggle
Table of con

downscale = True

output_path = prepare_experiment(Path('/kaggle/working'))

model = SuperTrackingModel(batch_s, stack_s, out_path=output_path, downscale=downscale)

train_gen = DataGenerator(Path('../input/tennisttrackingassignment/train/'), [1, 2, 3, 4, 5, 6])
val_gen = DataGenerator(Path('../input/tennisttrackingassignment/test/'), [1, 2], stack_s=stack_s)

model.train(train_gen.random_g, val_gen.random_g)

Starting training...
Epoch 1/20
150/150 0s 99ms/step - io_u: 0.0046 - loss: 1.1256
Epoch 1: val_loss improved from inf to 1.04910, saving model to /kaggle/working/exp_1/model_weights_best.weights.h5
150/150 49s 142ms/step - io_u: 0.0047 - loss: 1.1254 - val_io_u: 0.0017 - val_loss: 1.0491
Epoch 2/20
150/150 0s 99ms/step - io_u: 0.0191 - loss: 1.0245
Epoch 2: val_loss improved from 1.04910 to 0.90067, saving model to /kaggle/working/exp_1/model_weights_best.weights.h5
150/150 17s 113ms/step - io_u: 0.0192 - loss: 1.0245 - val_io_u: 0.1313 - val_loss: 0.9007
Epoch 3/20
150/150 0s 98ms/step - io_u: 0.0806 - loss: 0.9508
```

```
File Edit View Run Settings Add-ons Help
+ - ✂ 📄 📁 ▶ ⏏ Run All Code
Draft Session off (run a cell to start)
Notebook
Input
+ Add Input
DATASETS
> tennist
Output
> /kaggle
Table of con

s: 0.7718
Epoch 16/20
150/150 0s 99ms/step - io_u: 0.3597 - loss: 0.6706
Epoch 16: val_loss did not improve from 0.65131
150/150 16s 107ms/step - io_u: 0.3598 - loss: 0.6706 - val_io_u: 0.3180 - val_loss: 0.7122
Epoch 17/20
150/150 0s 98ms/step - io_u: 0.3847 - loss: 0.6453
Epoch 17: val_loss did not improve from 0.65131
150/150 16s 107ms/step - io_u: 0.3846 - loss: 0.6454 - val_io_u: 0.2140 - val_loss: 0.8159
Epoch 18/20
150/150 0s 99ms/step - io_u: 0.3597 - loss: 0.6702
Epoch 18: val_loss improved from 0.65131 to 0.64677, saving model to /kaggle/working/exp_1/model_weights_best.weights.h5
150/150 16s 108ms/step - io_u: 0.3597 - loss: 0.6702 - val_io_u: 0.3832 - val_loss: 0.6468
Epoch 19/20
150/150 0s 99ms/step - io_u: 0.3481 - loss: 0.6817
Epoch 19: val_loss did not improve from 0.64677
150/150 16s 107ms/step - io_u: 0.3482 - loss: 0.6817 - val_io_u: 0.2228 - val_loss: 0.8071
Epoch 20/20
150/150 0s 118ms/step - io_u: 0.3763 - loss: 0.6535
Epoch 20: val_loss did not improve from 0.64677
150/150 19s 127ms/step - io_u: 0.3762 - loss: 0.6536 - val_io_u: 0.1210 - val_loss: 0.9085
Training completed.
```

```
File Edit View Run Settings Add-ons Help
+ - ✂ 📄 📁 ▶ ⏏ Run All Code
Draft Session off (run a cell to start)
Notebook
Input
+ Add Input
DATASETS
> tennist
Output
> /kaggle
Table of con

Running stub for loading model ...
Downloading...
From: https://drive.google.com/uc?id=1el6uwAe43q4pnZeDHtUtsj0Jy565P23N
To: /kaggle/working/loaded_weights.h5
100%|██████████| 2.58M/2.58M [00:00<00:00, 173MB/s]
Loading model done.
loading clip data (game 1, clip 1) downsampled
loading clip data (game 1, clip 1)
loading clip labels (game 1, clip 1)
doing predictions
1/1 1s 891ms/step
1/1 0s 31ms/step
1/1 0s 31ms/step
1/1 0s 23ms/step
1/1 0s 31ms/step
1/1 0s 23ms/step
1/1 0s 31ms/step
1/1 0s 23ms/step
1/1 0s 32ms/step
1/1 0s 23ms/step
1/1 0s 32ms/step
1/1 0s 23ms/step
1/1 0s 23ms/step
1/1 0s 31ms/step
1/1 0s 23ms/step
1/1 0s 23ms/step
```