```python
from typing import Tuple

import numpy as np


def gaussian_mixture_moments(
    w: np.ndarray,  # the mixture weights shape=(N,)
    mean: np.ndarray,  # the mixture means shape(N, n)
    cov: np.ndarray,  # the mixture covariances shape (N, n, n)
) -> Tuple[
    np.ndarray, np.ndarray
]:  # the mean and covariance of of the mixture shapes ((n,), (n, n))
    """Calculate the first two moments of a Gaussian mixture"""

    # mean
    mean_bar = np.average(mean, axis =0, weights=w)  # TODO: hint np.average using axis and weights

    # covariance
    # # internal covariance
    cov_int = np.average(cov, axis=0, weights=w)

    # # spread of means
    # Optional calc: mean_diff =
    mean_diff = mean-mean_bar

    cov_ext = np.zeros(shape=cov[0,:,:].shape)
    M = np.size(mean, axis=0)
    for i in range(0, M):
        cov_ext += w[i]*(mean_diff[i, ...]).T@(mean_diff[i, ...])
    # # total covariance
    cov_bar = cov_int + cov_ext

    return mean_bar, cov_bar


def test_gaussian_mixture_moments():
    w = np.array([0.5, 0.5])
    mean = np.array([[5.0, 10.0], [3, 4]])
    cov = np.array([np.eye(2), np.eye(2)])

    mean_bar, cov_bar = gaussian_mixture_moments(w, mean, cov)
    print(mean_bar)
    print(cov_bar)

test_gaussian_mixture_moments()
```