File - performance_test

```
C:\Users\usuario\Anaconda3\envs\tfq\python.exe C:/Users/usuario/qGAN/quantumGAN/performance_test.py
[0.78824747 0.00912633 0.54242675 0.11601135 0.23433377 0.69136728]
Epoch 0: Loss: [-0.48500819] [0.4948152  0.         0.44553522 0.        ] [0.42578125 0.10058594 0.43066406 0.04296875]
[0.12114191] [0.11551757]
Timer unit: 1e-07 s

Total time: 27.4484 s
File: C:\Users\usuario\qGAN\quantumGAN\quantum_generator.py
Function: train_mini_batch at line 129

Line #      Hits         Time  Per Hit   % Time  Line Contents
==============================================================
   129                                           def train_mini_batch(self):
   130         2        171.0     85.5      0.0         nabla_theta = np.zeros(self.parameter_values.shape)
   131         2         27.0     13.5      0.0         new_images = []
   132
   133        17        764.0     44.9      0.0         for _, noise in self.mini_batch:
   134       105       2326.0     22.2      0.0             for index in range(len(self.parameter_values)):
   135        90       7157.0     79.5      0.0                 perturbation_vector = np.zeros(len(self.parameter_values
))
   136        90       2160.0     24.0      0.0                 perturbation_vector[index] = 1
   137
   138        90       8227.0     91.4      0.0                 pos_params = self.parameter_values + (np.pi / 4) *
perturbation_vector
   139        90       5130.0     57.0      0.0                 neg_params = self.parameter_values - (np.pi / 4) *
perturbation_vector
   140
   141        90  127697936.0 1418866.0     46.5                 pos_result = self.get_output(noise, params=pos_params)
   142        90  125644022.0 1396044.7     45.8                 neg_result = self.get_output(noise, params=neg_params)
   143
   144        90      90141.0   1001.6      0.0                 pos_result = self.discriminator.predict(pos_result)
   145        90      42405.0    471.2      0.0                 neg_result = self.discriminator.predict(neg_result)
   146        90      41000.0    455.6      0.0                 gradient = self.BCE(pos_result, np.array([1.])) - self.
BCE(neg_result, np.array([1.]))
   147        90      17175.0    190.8      0.0                 nabla_theta[index] += gradient
   148        15   20924527.0 1394968.5      7.6             new_images.append(self.get_output(noise))
   149
   150        14        212.0     15.1      0.0         for index in range(len(self.parameter_values)):
   151        12        456.0     38.0      0.0             self.parameter_values[index] -= (self.learning_rate / self.
mini_batch_size) * nabla_theta[index]
   152
   153         2        227.0    113.5      0.0         self.mini_batch = [(datapoint[0], fake_image) for datapoint,
fake_image in zip(self.mini_batch, new_images)]


Process finished with exit code 0

```