Quantum Imaging

Kshitij Aphale (B21CS089)
Krishna Gaurang (B21EE086)

INTRODUCTION

Quantum Imaging: Exploits the quantum correlations between particles to improve certain aspects of imaging beyond what's classically possible (ex: super resolution).

Types of Quantum Imaging: Ghost Imaging, Interaction free Measurement,sub-shot-noise imaging.

Steps Involved:

1. Create Entanglement between photons.
2. Measure quantum correlations and make observations based on type of imaging.
3. Convert the observed data into a raw image.
4. Clean the image using image processing techniques to obtain a final image.

Quantum States and Gates

In quantum mechanics, properties of particles are not quantified as numbers but rather described as having various probabilities of being different numbers, denoted as $|\psi\rangle$. Each state is further described as being in a superposition of certain basis states $|\psi\rangle$=a$|\psi_1\rangle$ +b$|\psi_2\rangle$+c$|\psi_3\rangle$. Here a system in state $|\psi\rangle$ has probabilities a^2,b^2, and c^2 respectively of being in basis state $|\psi_1\rangle$,$|\psi_2\rangle$, or $|\psi_3\rangle$.

A Qubit is a system which has 2 basis states (let us define them as $|0\rangle$ and $|1\rangle$). The qubit's state is defined as superposition of these states a$|0\rangle$ + b$|1\rangle$. For convenience it is represented in vector notation as $(a\ \ b)^T$.

A quantum gate is an interaction which changes the state of the qubit. They are represented by matrices, and the new state is just the product of the matrix with the current state vector.

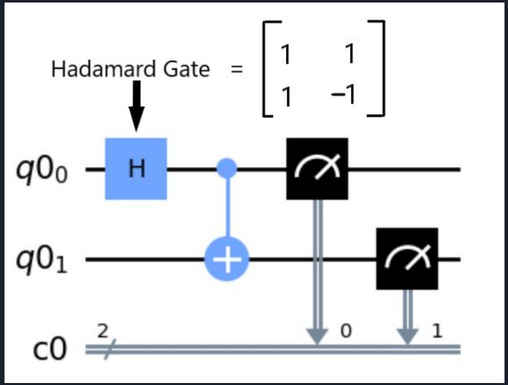
1. Creating Entanglement

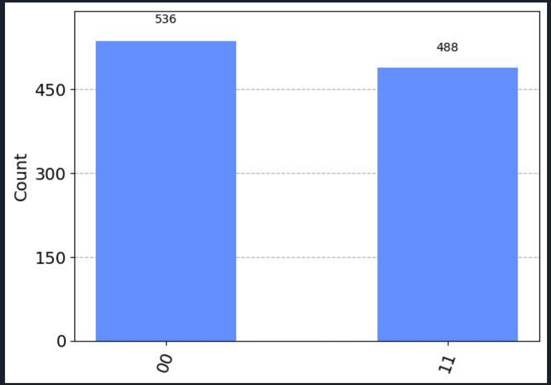
The state of 2 qubits is represented as $|\psi_a\rangle^{\otimes}|\psi_b\rangle$ (Tensor product). 2 Particles are said to be entangled when their states cannot be represented as a tensor product of individual states ie- They act like a single system.

The most famous entangled states are the bell states and they are generated by applying a Hadamard gate followed by a CNOT gate.

We can simulate that using some python code and the Qiskit library as shown below

```python
# Creating Registers
qr=QuantumRegister(2)
cr=ClassicalRegister(2)
circuit=QuantumCircuit(qr,cr)

# Creating Hadarmard and CNOT gate and Making measurement
circuit.h(qr[0])
circuit.cx(qr[0],qr[1])
circuit.measure(qr,[cr[0],cr[1]])

# Simulating the experiment
simulator=Aer.get_backend('qasm_simulator')
result=execute(circuit,backend=simulator).result()
```





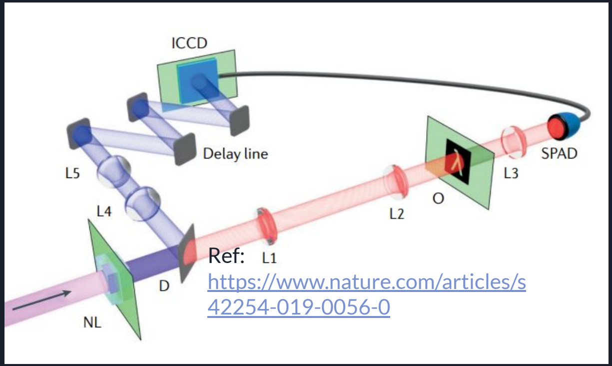
2. Measuring Correlations and Observations

In ghost imaging, one arms contain single photon detector (the one with the object) and the other spatially resolving sensor. Correlations between the 2 signals are used to recreate an image.

We can simulate this classically in python using the pixel values as absorptivity values, using them as probabilities that the photon will pass through. The errors can be simulated using random number generator to simulate spatial correlation errors, and background noise.

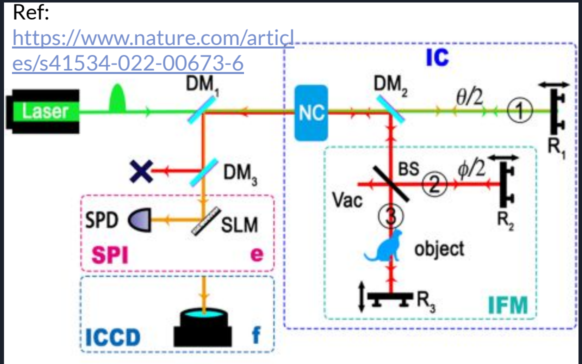
In Interaction free Measurement we make use of an interferometer to observe constructive and destructive interference of signal and idler beams, and observe how they differ in the presence vs absence of an object.

GGhost Imaging Setup



Ref: https://www.nature.com/articles/s42254-019-0056-0

Interaction Free Measurement setup



Ref: https://www.nature.com/articles/s41534-022-00673-6

3. Converting Observed Data into Raw Images

The Ghost Imaging Simulation, yields the following example on an example object of a digit 2 and an example from a research paper.

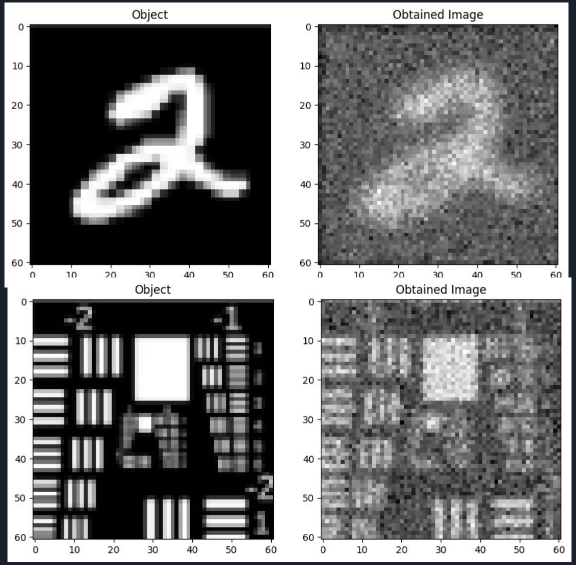
Defining Errors

```python
# Loading the image in black and white
fig=plt.figure(figsize=(10,10))
obj1=img.imread('/content/drive/MyDrive/quantum_imaging/mnist_2_new.png')
obj=obj1[:,:,0]
fig.add_subplot(1,2,1)
plt.imshow(obj[1:-2,1:-2],cmap='Greys')
plt.title('Object')

# The size of our camera sensor
frame_size=64
# Note: The object matrix contains absorptivity values for the object
photons_per_pixel=100
bucket_detector=np.zeros(frame_size**2)
single_photon_detector=np.zeros((frame_size,frame_size))

# Defining scale of all errors
error_in_spacial_corr=0.03 # out of 1
surrounding_photons_prob = 0.01 # out of 1
error_in_apparatus = 0.05 # out of 1
```

Simulation

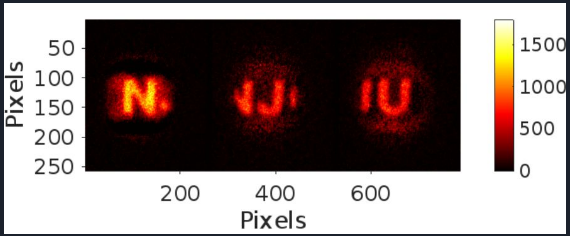
```python
# Simulating the experiment
c=0
for k in range(photons_per_pixel):
  for i in range(frame_size):
    for j in range(frame_size): # Simulating error in spacial correlation
      random_number=np.random.randn(1)+0.5
      random_number_for_noise=np.random.randn(1)+0.5
      random_number_for_error_in_apparatus=np.random.randn(1)+0.5
      # Bucket Detector
      if random_number<=obj[i,j]:
        bucket_detector[c]+=1
      if random_number_for_noise <= surrounding_photons_prob:
        if random_number_for_error_in_apparatus > error_in_apparatus:
          bucket_detector[c]+=1
      # Spatial Detector
      if random_number>obj[i,j]:
        error_x=error_in_spacial_corr*frame_size*random.uniform(-2,2)
        error_y=error_in_spacial_corr*frame_size*random.uniform(-2,2)
        X=round(i+error_x)
        Y=round(j+error_y)
        if X>=frame_size:
          X=frame_size-1
        elif X<0:
          X=0
        if Y>=frame_size:
          Y=frame_size-1
        elif Y<0:
          Y=0
        single_photon_detector[X,Y]+=1
  c+=1
```

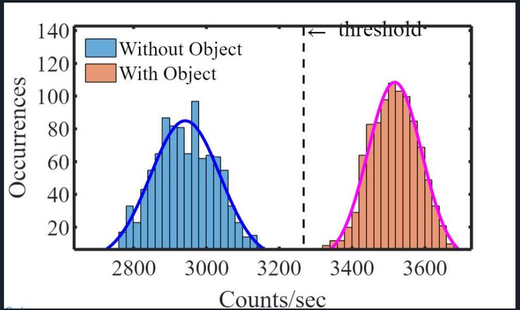
Results



3. Converting Observed Data into Raw Images

Interaction free measurement gives us counts for interference with and without object. Which is then used to construct the images. Using the matlab code from the following paper we can observe the IFM Results.https://github.com/YiquanYang/Interaction-free-single-pixel-quantum-imaging-with-undetected-photons

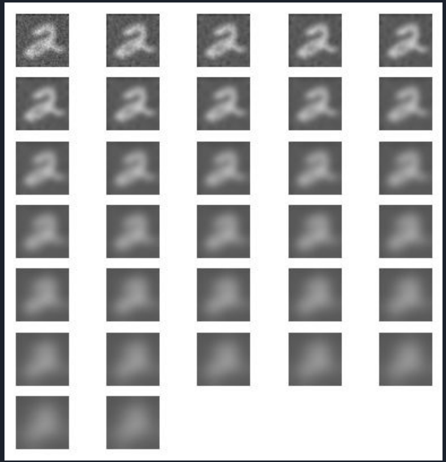




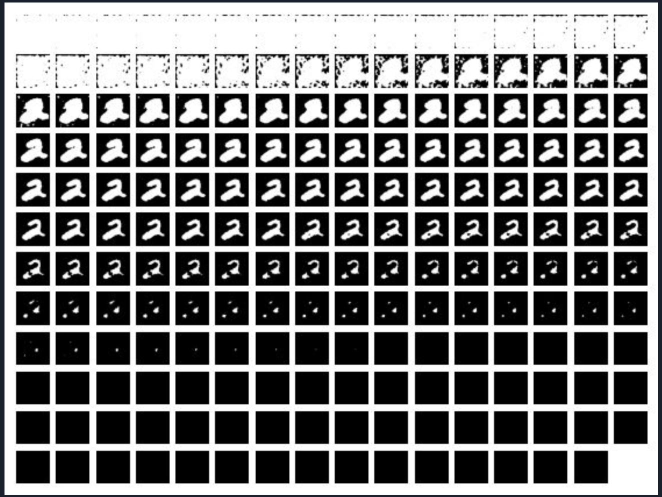
4. Clean the Image

We can use image processing techniques such as Gaussian filter and thresholding to get rid of most of the noise and get a nice and crisp image. This technique applied to the ghost imaging example show before is shown below (using opencv library in python).

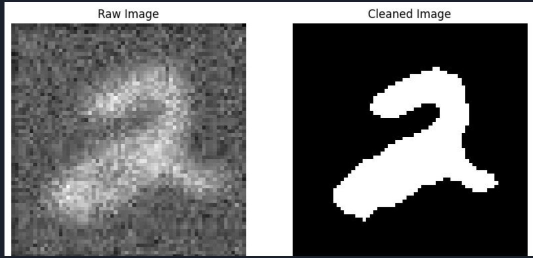
Try various blur amounts



Try various Threshold values



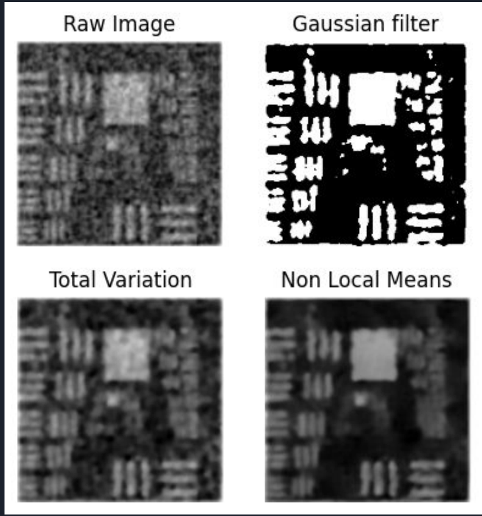
Apply best values



4. Clean the Image

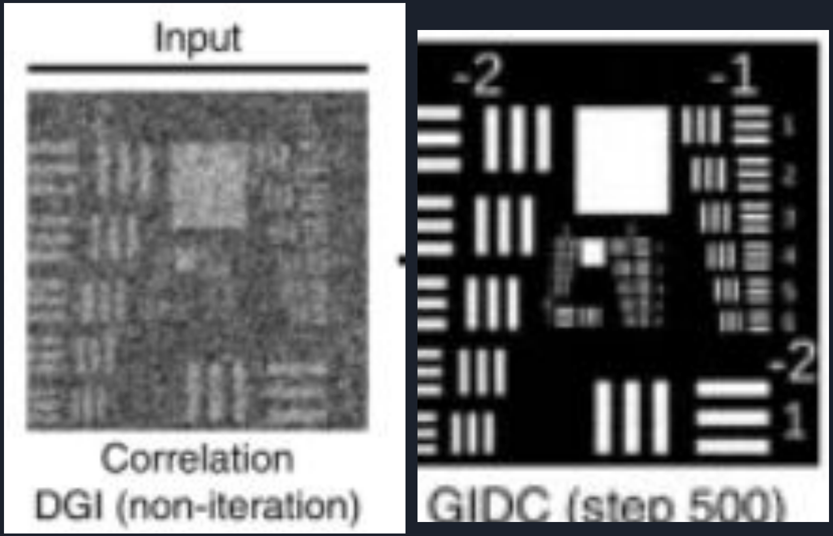
Other advanced techniques like Deep Neural network constraint conducted by experts produce an even better image. A possible way to create a Deep Learning model using CNN, is to simulate ghost imaging (as shown previously) on a large set of images, and using this data of noisy vs clean image, train a CNN model to clean any future ghost imaging images.

Using various Cleaning Techniques



Experiments Using GIDC

Ref: https://github.com/FeiWang0824/GIDC



Applications of Quantum Imaging

Quantum Imaging is currently in its early stage. Once developed, it can find applications in the following areas:

- Secured Imaging: We already know how useful quantum secured encryption is, even for imaging purposes (ex: radar) classically poses a risk. Photons can be captured, altered and sent back. To detect altering of photons (thus altering of image), they can be entangled with certain other photons (not to be sent), and on receiving back the signal, correlations can be compared to detect altering. (ref: https://pubs.aip.org/aip/apl/article/101/24/241103/125052/Quantum-secured-imaging).
- Sensing: Quantum imaging can be used in sensing applications, such as detecting weak signals or measuring very small changes in physical systems.
- Ghost imaging for remote sensing: Ghost imaging can be used for remote sensing, such as in LiDAR applications, to generate 3D images of objects at a distance.
- Quantum-enhanced microscopy: Quantum imaging techniques can potentially improve the resolution and sensitivity of microscopy, enabling the study of biological and physical systems at the quantum level.

THANK YOU