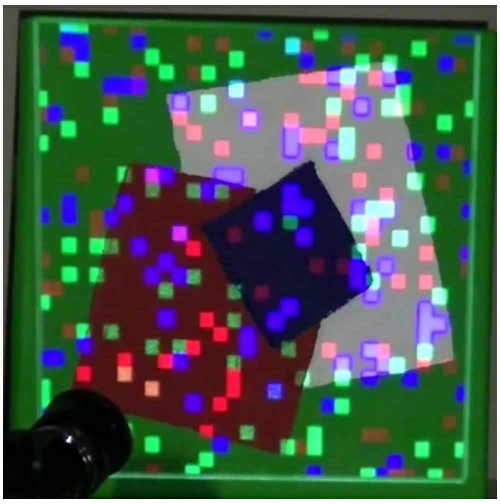
Compressive Ghost Imaging

How to do it yourself



Authors: Bas Goorden and Jin Lian, Complex Photonic Systems (COPS),
University of Twente

2015

Contents

1. What is compressive ghost imaging?

Every day, we use our eyes and cameras to take images of objects and scenes. Eyes as well as cameras consist of a very large number of detectors, in the case of cameras usually called pixels. In both cases, there is a lens in front of the detectors, which guides the light from each point on the object to a different detector. By recording the amount of light reaching each detector we know the amount of light scattered from each point on the object, providing us with a complete image of the object.

Compressive ghost imaging takes a radically different approach. We will use only a single detector, which will measure the total amount of light coming from an object. Of course there are many different objects that scatter the same total amount of light. Therefore, we measure how the total amount of light coming from the object changes *as we change the shape of the light that we shine onto the object*.

Example 1. We shine light from behind the object, such that the object blocks the light. If the shape of the light is equal to the shape of the object, all the light will be blocked and no light will be measured by our single pixel detector. Conversely, if the shape of the light is such that all the light misses the object, a lot of light will be measured.

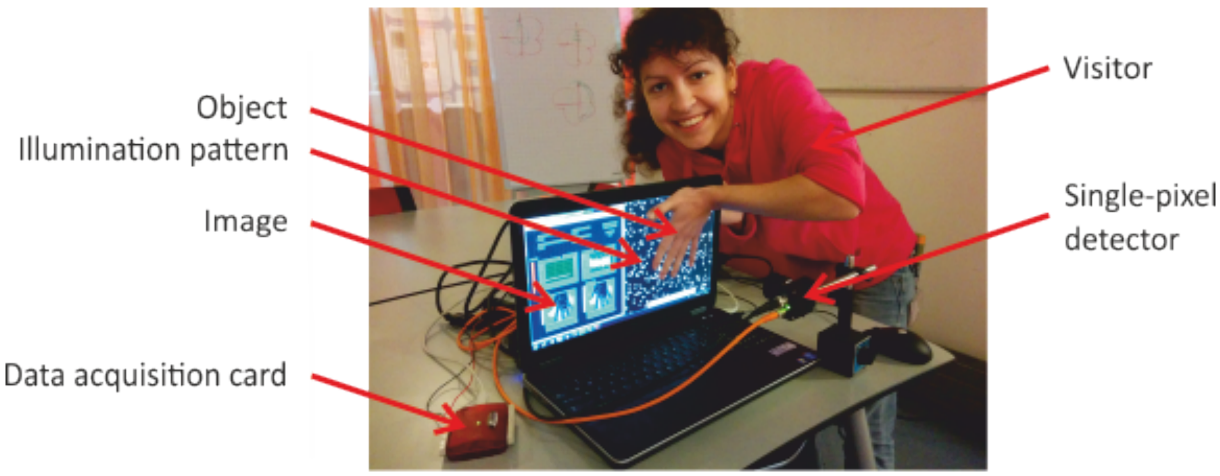


Fig. 1, Compressive ghost imaging using the laptop screen as an illumination device. In this case, the object is a human hand. The laptop displays a sequence of illumination patterns and the single-pixel detector measures the total transmitted intensity. The data acquisition card is used to transfer the data to the laptop, which reconstructs the image and displays it on the screen.

Example 2. If we shine the light from the front of the object, such that the object reflects the light to the detector, it is exactly the other way around. If the shape of the light is equal to the shape of the object, the amount of light measured is maximum. If the light shape is completely different from the object, the amount of light measured is zero.

Since we do not know what the object looks like before taking the image, we choose the light shapes that we shine onto the object randomly. For every random light shape, the total amount of light measured tells us how similar the object is to this light shape. This way, if we use enough different light shapes, we can find the exact shape of the object.

In principle, the number of light shapes required is equal to the number of pixels in the image we want to take. The reason is that if less light shapes are used, there exist multiple objects that would give exactly the same measurement results on the single pixel detector and it seems impossible to decide which of these “allowed” objects is the correct one.

To solve this, we use a very similar trick to the one that allows the strong reduction of normal images to a very small file size, using for example JPEG compression. The trick is that most objects are sparse, which means that the points on a real life object are not random: they are in some way related to each other. We will assume that our objects can be built from a relatively small number of square blocks of various sizes.

In the compressive ghost imaging experiment that we will build, we will use typically 3 to 10 times less different light shapes than there are pixels in the image we want to reconstruct. This means that our measurement result will lead to a very big pool of different objects that are allowed. Then, we will start a calculation, which searches and finds out of this huge number of possible objects the one that can be built from the smallest number of squares, where the squares are allowed to have different sizes and different positions. As it turns out, this calculation finds the true object and therefore we have made an image of the object! And this by doing a number of measurements much smaller than what is normally required, effectively compressing the image *while taking it*!

A nice tutorial on compressive sensing can be found here: http://www.brainshark.com/brainshark/brainshark.net/portal/title.aspx?pid=zCdz10BfTRz0z0

An example of the experiment described in this document can be viewed in a short YouTube movie: https://www.youtube.com/watch?v=AVv6bkmFpVc

2. Setting up the hardware...

Hardware requirements:

- Computer (desktop or laptop, faster is better)
- A photodiode (single pixel light detector) with an electrical amplifier; suggested model: Thorlabs PDA100A Si Switchable Gain Detector.
- An analog-to-digital converter. We suggest to use a Labview programmable USB Data Acquisition Card; suggested model: RedLab 1608FS

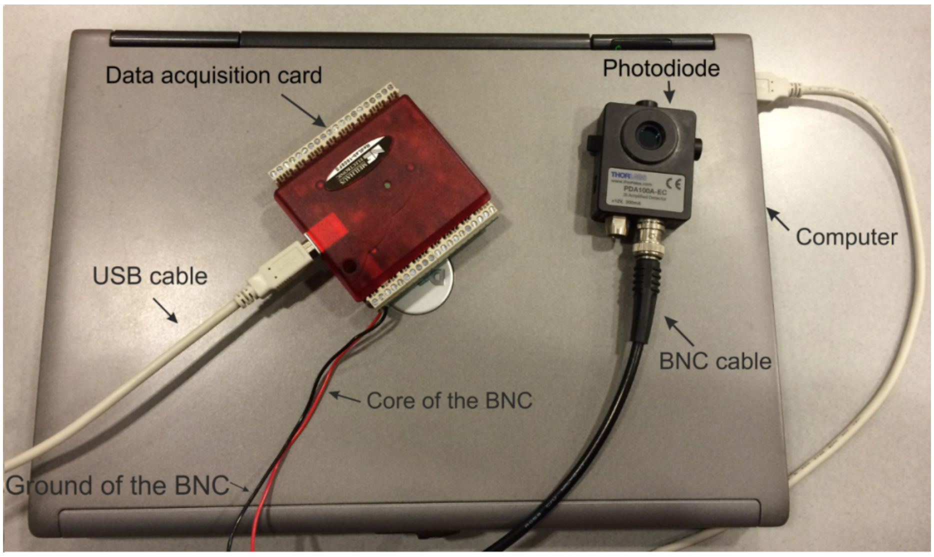


Fig. 2, Required hardware: a computer, a DAQ card and a photodiode.

Optional hardware:

- Projector; suggested model: Optoma PK320
- Lens which increases the efficiency of collecting signal.

Fig. 3, Compressive ghost imaging setup, including the optional projector and lens. The DAQ card is hidden from view. In this setup, the projector illuminates the object. In this case the object is a painting. The photodiode with the lens, positioned near the object, collects light scattered from the object. The data acquisition card, here hidden from view, is used to transfer the data to the laptop, which reconstructs the image and displays it on the screen.

To detect the light signal, a photodiode (single pixel detector) is the simplest device which can be used. It converts light signal to electric current signal. By connecting it to a resistor, the current signal can be converted to the voltage signal. For some photodiodes, they have built-in electric amplifier which can amplify the generated electric signal.

For reading and processing the signal, an analog-to-digital converter is needed. Most of the cases in the lab, this is done by data acquisition cards (DAQ card). The DAQ card converts the analog signal to digital signal, and it is programmable by the computer through the USB connection. It reads signal by a BNC (coaxial cable) cable from the photodiode, converts it to digital signal and sends it to the computer by the USB cable which is also used for programming the DAQ card.

Setting up the hardware :

1. Connect the DAQ card to the computer by USB.
2. Power up the photodiode and connect it to the DAQ card by BNC cable. Sometimes the DAQ card does not have a BNC plug for BNC connector,

so you have to connect the core of the BNC to the analog pin of the DAQ card, the ground of the BNC (shell) to the ground pin of the DAQ card.

3. (optional) Mount the lens onto the photodiode so that as much light as possible is collected from the object onto the photodiode.
4. (optional) Connect the projector to the computer, using the VGA cable. Power up the projector and set the computer to extend the screen onto the projector, which acts as a second screen.
5. Place the object in front of the illumination device (either directly in front of the screen or 0.5-1m in front of the projector).
6. Place the photodiode 20-30 cm from the object, with the detector toward the object.

3. The software…

The software has three functions:

1. Write illumination patterns for display on the illumination device;
2. Read photodiode measurement data;
3. Run a sparse optimization algorithm to reconstruct the correct image.

Any software that can perform these tasks can be used for the experiment. Our suggested software is:

1. Labview (for tasks 1 and 2), combined with
2. Matlab (for task 3).

The code that we used is available online at
https://sourceforge.net/projects/compressiveghostimaging/

There are three versions:

1. CompressiveGhostImaging.vi
   This is Labview software, which writes colored patterns to a second screen (typically, a projector). It records the corresponding total intensities measured by the photodiode. Finally, it starts Matlab and runs a sparse optimization algorithm (ASP, by M. Friedlander and M. Saunders, http://www.cs.ubc.ca/~mpf/asp/) to reconstruct the image.

The algorithm takes as input the illumination patterns and the measurement data and outputs the image. The image is displayed on the screen.
2. CompressiveGhostImaging_blackAndWhite.vi
   This program does the same, but sends only black and white patterns. We use this program when we use the laptop screen as illumination device.
3. CompressiveGhostImaging_blackAndWhite_NIDAQ.vi
   This program does the same as 2., but implements a National Instruments DAQ card instead of the suggested Redlab DAQ card.

All versions of the Labview software require setting the path to the “asp” folder, which contains the optimization routines.

4. And doing the experiment!

After setting up the hardware and starting the program, you are ready to go. Running the program with its default settings should create your first compressive ghost image!

5. FAQ:

It does not work, what should I do?

1. Change the position of the detector (and, if present, the lens) to maximize the amount of light that reaches the detector (and comes from the object).
2. Verify that the intensity measured by the detector changes as different illumination patterns are displayed.
3. Decrease background light and other potential sources of noise.
4. Increase the delay time (between displaying the illumination pattern and recording the intensity) to make sure the problem is not lag of the display device.
5. If the previous steps do not work, choose a simpler object and/or
6. Increase the number of illumination patterns.