Chapter 1

Introduction

It has long been the goal of computer vision researchers to be able to develop systems that can reliably recognize objects in a scene. Achieving this unlocks a huge range of applications that can benefit society as a whole. From fully autonomous vehicles, to automatic labelling of uploaded videos/images for searching, or facial recognition for identification and security, the uses are far reaching and extremely valuable. The challenge does not lie in finding the right application, but in the difficulty of training a computer to *see*.

There are nuisance variables such as changes in lighting condition, changes in viewpoint and background clutter that do not affect the scene but drastically change the pixel representation of it. Humans, even at early stages of their lives, have little difficulty filtering these out and extracting the necessary amount of information from a scene. So to design a robust system, it makes sense to design it off how *our* brains see.

Unfortunately, vision is a particularly complex system to understand. It has more to it than the simply collecting photons in the eye. An excerpt from a recent Neurology paper [1] sums up the problem well:

> It might surprise some to learn that visual information is significantly degraded as it passes from the eye to the visual cortex. Thus, of the unlimited information available from the environment, only about 10^{10} bits/sec are deposited in the retina ... only $\sim 6 \times 10^6$ bits/sec leave the retina and only 10^4 make it to layer IV of V1 [2], [3]. These data clearly leave the impression that visual cortex receives an impoverished representation of the world ... it should be noted that estimates of the bandwidth of conscious awareness itself (i.e., what we 'see') are in the range of 100 bits/sec or less[2], [3].

Current digital cameras somewhat act as a combination of the first and second stage of this system, collecting photons in photosensitive sensors and then converting this to an image on the order of magnitude of 10^6 pixels (slightly larger but comparable to the 10^6 bits/sec travelling through the optic nerve).

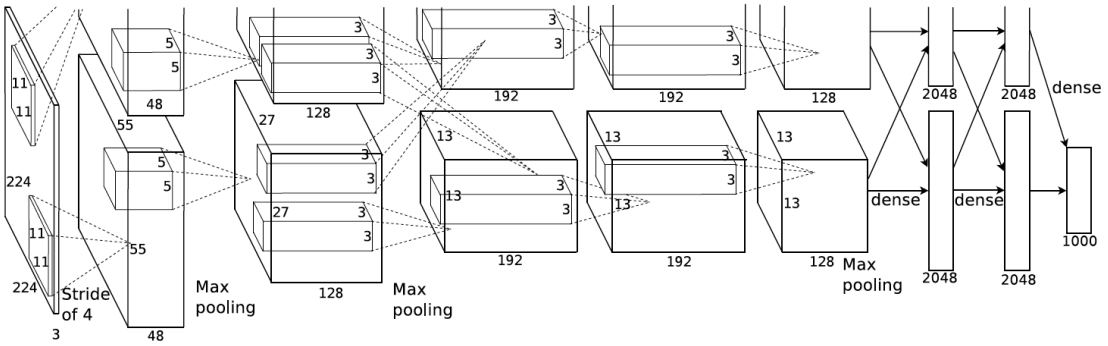


Figure 1.1: **Convolutional Architecture of [8].**

If we are to build effective vision systems, it makes sense to emulate this compression of information. The question now stands before us — what information is kept on entry to the V1 cortex? Hubel and Wiesel revolutionized our understanding of the V1 cortex in the 50s and 60s by studying cats [4], [5], macaques and spider monkeys [6]. They found that neurons in the V1 cortex fired most strongly when edges of a particular (i.e., neuron-dependent) orientation were presented to the animal, so long as the edge was inside the receptive field of this neuron. Continued work on their experiments by Blakemore and Cooper [7] showed, by exposing kittens to controlled environments in which they only saw horizontal and vertical lines, that these early layers of perception are in fact *learned*.

1.1 Convolutional Neural Networks

The current state of the art in image understanding sysems are Convolutional Neural Networks (CNNs). These are a learned model that stacks many convolutional filters on top of each other separated by nonlinearities. They are seemingly inspired by the visual cortex in the way that they are hierarchically connected, progressively compressing the information into a richer representation.

Figure 1.1 shows an example architecture for the famous AlexNet [8]. Inputs are resized to a manageable size, in this case 224×224 pixels. Then multiple convolutional filters of size 11×11 are convolved over this input to give 96 output *channels* (or *activation maps*). In the figure, these are split onto two graphics cards or GPUs for memory purposes. These are then passed through a a pointwise nonlinear function, or a *nonlinearity*. The activations are then pooled (a form of downsampling) and convolved with more filters to give 256 new channels at the second stage. This is repeated 3 more times until the 13×13 output with 256 channels is unravelled and passed through a fully connected neural network to classify the image as one of 1000 possible classes.

CNNs have garnered lots of attention since 2012 when the previously mentioned AlexNet nearly halved the top-5 classification error rate (from 26% to 16%) on the ImageNet Large Scale Visual Recognition Competition (ILSVRC) [9][1]. In the years since then, their complexity has grown significantly. AlexNet had only 5 conovlutional layers, whereas the 2015 ILSVRC winner ResNet [15] achieved 3.57% top-5 error with 151 convolutional layers (and had some experiments with 1000 layer networks).

1.2 Problems with CNNs and Project Motivation

Despite their success, they are often criticized for being *black box* methods. You can view the first layer of filters quite easily (see Figure 1.2a) as they exist in RGB space, but beyond that things get trickier as the filters have a third, *depth* dimension typically much larger than its two spatial dimensions. Additionally, it is not clear what the input channels themselves correspond to. For illustration purposes, we have also shown some example activations from the first three convolutional layers for AlexNet in Figure 1.2b-Figure 1.2d. These activations are taken after a specific nonlinearity that sets negative values to 0 (hence the large black regions). We can see in 'conv1' (Figure 1.2b) that some of the first layer channels are responding to edges or colour information, but as we go *deeper*, it becomes less and less clear what each activation is responding to.

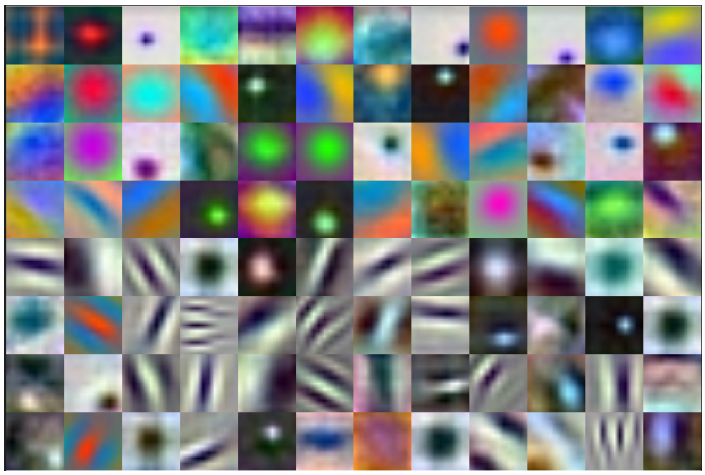
Aside from their lack of interpretability, it takes a long time and a lot of effort to train state of the art CNNs. Typical networks that have won ILSVRC since 2012 have had roughly 100 million parameters and take up to a week to train. This is optimistic and assumes that you already know the necessary optimization or architecture hyperparameters, which you often have to find out by trial and error. In a conversation the author had with Yann LeCun, the attributed father of CNNs, at a Computer Vision Summer School (ICVSS), LeCun highlighted this problem himself:

> There are certain recipes (for building CNNs) that work and certain recipes that don't, and we don't know why.

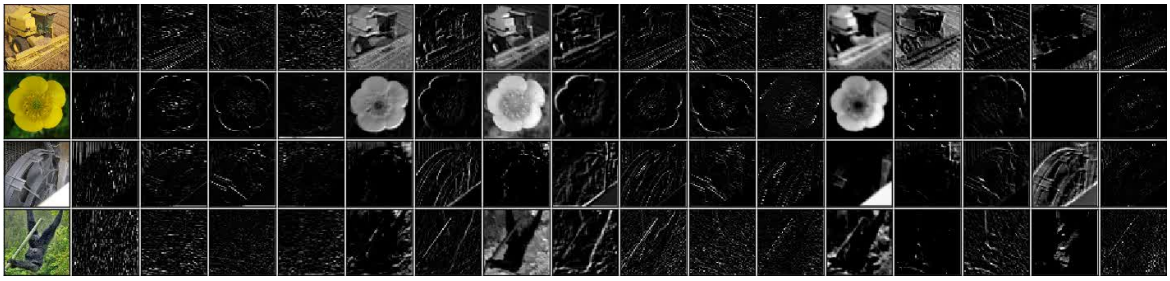
Considering the recent success of CNNs, it is becoming more and more important to understand *how* and *what* a network learns, which has contributed to it making its classification or regression choice. Without this information, the use of these incredibly powerful tools could be restricted to research and proprietary applications.

They are fairly crude in terms of signal processing, as the filters are arbitrary FIR filters. It is important to find ways to find an adequate richness to the filtering. It must still be large yet the number of the parameters needed to specify this should be kept as small as possible.

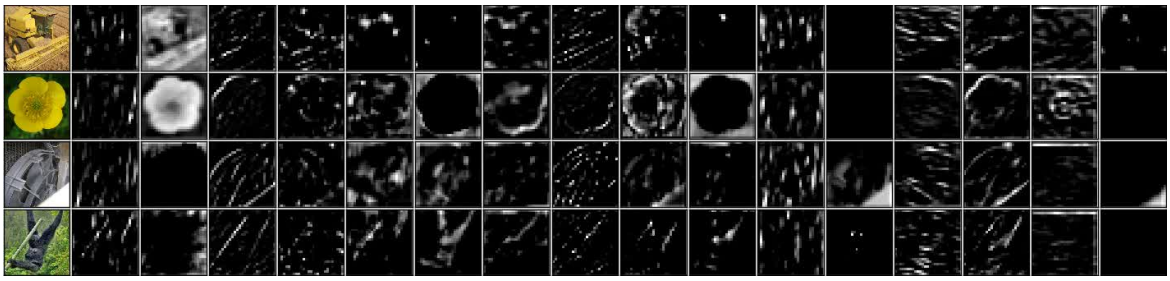
[1] The previous state of the art classifiers had been built by combining keypoint extractors like SIFT[10] and HOG[11] with classifiers such as Support Vector Machines[12] and Fisher Vectors[13], for example [14].



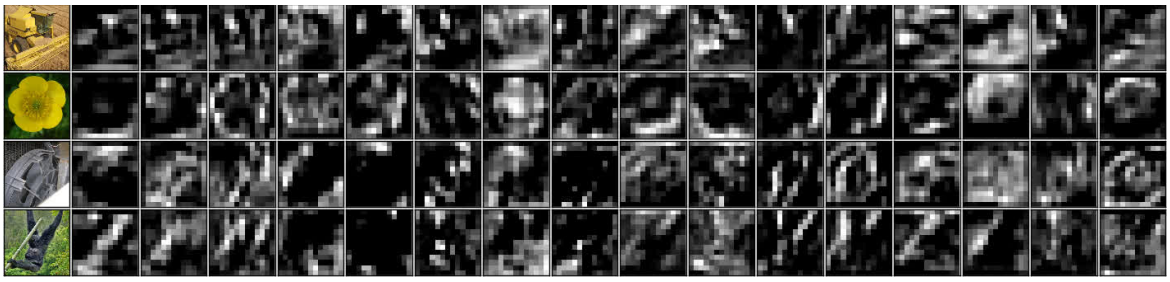
(a) conv1 filters



(b) conv1 activations



(c) conv2 activations



(d) conv3 activations

Figure 1.2: **The first layer filters learned by AlexNet and the first three layer's activations.** (a) The 11×11 filters for the first stage of AlexNet. Of the 96 filters, 48 were learned on one GPU and another 48 on another GPU. Interestingly, one GPU has learned mostly lowpass/colour filters and the other has learned oriented bandpass filters. (b) - (d) Randomly chosen activations from the output of the first, second and third convolutional layers of AlexNet (see Figure 1.1) with negative values set to 0. Filters and activation images taken from supplementary material of [8].

In particular, the starting point for our project is clear — the filters learned by the first layer of a CNN, shown in **??**, look like oriented wavelets. This is not surprising either from a biological point of view — as it matches the earlier mentioned results by Hubel and Wiesel, or from a signal processing point of view — as the wavelet transform is a powerful and stable way to split up an image into areas of the frequency domain.

With this, the above goal can then be refined to a more targeted one:

Starting with a wavelet transform, we want to look at how to develop a higher order system by drawing inspiration from the second and third layers of a CNN. Achieving this will shed light on what is learned and what is needed for a good image recognition system.

The second inspiration for a starting point is the recent work in developing *Scatternets* by Stephané Mallat, one of the forefathers of the wavelet transform, and his research group. Their work attempts to do precisely what we want — start with a wavelet transform that is sensitive to edges, and then build deeper layers on top of this that are sensitive to larger shapes, while being insensitive to uninformative variations such as translation, rotation, and scale. Their Scatternets are purely deterministic.

For example the evolution of the convolutional net over the MLP is a classic example of how restriction led to improvement.

As an example[2], it is not hard to imagine a deep network that could be used to assess whether giving a bank loan to an applicant is a safe investment. It could compare a vector of their current and past financial situation to a dataset of others and choose a simple yes/no answer. Trusting a black box solution is deeply unsatisfactory in this situation. Not only from the customer's perspective, who, if declined, has the right to know why [16], but also from the bank's — before lending large sums of money, most banks would like to know why the network has given the all clear. 'It has worked well before' is a poor rule to live by.

We are not the first research group to be unsatisfied with not knowing the mechanics of CNNs. There has been a lot of very impressive work done recently on trying to visualize the response a CNN has to a given input, in particular the work done by **zeiler_visualizing_compact_2014**, which builds 'Deconvolutional Neural Networks' (deconvnets) to view the regions of the input image that cause large responses at deeper layers of a CNN. This was the key tool in the previously mentioned 'Why Should I Trust You?' paper [17] to find the regions of the image that make up **??**.

So far these tools, while useful, stop short of turning visualization around into an improved strategy for designing networks. This brings us to the main goal of our research:

We hope to further research into understanding *how* and *what* deep networks learn by building a well understood and well-defined network that mimics their operation. Intuition is the primary goal, and with that, we believe improved performance will follow.

[2]While we limit the scope of our project to Imaging problems, CNNs can and have been used successfully in time series and language tasks.

We do not wish to focus only on purely handcrafted methods, nor on purely learned methods. Neither is discounted, and a hybrid of the two seems to be a good way to achieve our goal.

1.3 Layout

The layout of the report is as follows. **??** explores some of the background necessary for starting to develop image understanding models. In particular, it covers the inspiration for CNNs and the workings of CNNs themselves.

?? covers the wavelet transform used by Mallat, and compares it to the preferred Dual-Tree Complex Wavelet Transform (DTCWT) by Kingsbury [18].

?? reviews in depth the Scatternet designs by Mallat et. al. This is the last chapter of literature review, before **??** which starts to explore our work and analysis done on these Scatternets. In particular, we swap out the Morlet wavelets from Mallat's Scatternet to the faster, separable DTCWT wavelets, but we also make changes to the design of the Scatternet and look at how to apply the principles of visualizations, like those in the work by **zeiler_visualizing_compact_2014**.

?? then explores our recent work in attempting to combine Scatternets with CNNs. The inspiration for this being the idea that a well designed tool like the Scatternet, followed by a shallower CNN, should be equivalent to or better than a deeper CNN.

chapter 2 then summarizes our findings so far, discusses and analyses the results, and lays out the plan for the remainder of the project.

This work is stimulated by the intuition that wavelet decompositions, in particular complex wavelet transforms, are good building blocks for doing image recognition tasks. Their well understood and well defined behaviour as well as the similarities seen in learned networks, implies that there is potential gain for thinking about CNN layers in a new light.

To explore and test this intuition, we begin by looking at one of the most popular current uses of wavelets in image recognition tasks, in particular the Scattering Transform.

1.4 Series Expansions of Signals

Look at the intro to Vetterli's book. Want to make a statement about expanding signals in some form or another.

1.5 Contributions

The contributions and layout of this thesis are:

- **Software for wavelets and** DT$\mathbb{C}$WT **based ScatterNet (chapter 3)**

- **ScatterNet analysis and visualizations (chapter 4)**. Presented at MLSP2017, this chapter
- **Invariant Layer/Learnable ScatterNet (chapter 5)** Presenting at ICIP2019.
- **Learning convolutions in the wavelet domain (chapter 6)**.

1.5.1 Desirable Properties

Unlike CNNs introduced earlier which have little prior constraints (apart from the commonly used L_2 regularization), the scattering operator may be thought of as an operator S that imposes structural priors on learning by extracting features with manually chosen, desirable properties. The extracted features can be used In classical paradigms of image understanding, it makes sense to add these priors, but it remains yet to be shown that these help learning.

limit variability these properties areview on these properties are manually chosen with the ultimate goal of aiding image understanding.

Chapter 2

Conclusion

This chapter aims to logically tie together the results from the previous chapter, outlining what has been promising and what has not been, offering explanations as to why we think that is the case.

References

[1] M. E. Raichle, "Two views of brain function", eng, *Trends in Cognitive Sciences*, vol. 14, no. 4, pp. 180–190, Apr. 2010.

[2] C. H. Anderson, D. C. Van Essen, and B. A. Olshausen, "Directed Visual Attention and the Dynamic Control of Information Flow", en, in *Neurobiology of Attention*, Elsevier, 2005, pp. 11–17.

[3] Tor Nørretranders, *The User Illusion*. Viking, 1998.

[4] D. H. Hubel and T. N. Wiesel, "Receptive fields of single neurones in the cat's striate cortex", *The Journal of Physiology*, vol. 148, no. 3, pp. 574–591, Oct. 1959.

[5] ——, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex", eng, *The Journal of Physiology*, vol. 160, pp. 106–154, Jan. 1962.

[6] ——, "Receptive fields and functional architecture of monkey striate cortex", eng, *The Journal of Physiology*, vol. 195, no. 1, pp. 215–243, Mar. 1968.

[7] C. Blakemore and G. F. Cooper, "Development of the Brain depends on the Visual Environment", en, *Nature*, vol. 228, no. 5270, pp. 477–478, Oct. 1970.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", in *NIPS*, Curran Associates, Inc., 2012, pp. 1097–1105.

[9] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge", *arXiv:1409.0575 [cs]*, Sep. 2014. arXiv: 1409.0575 [cs].

[10] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints", *International Journal of Computer Vision*, vol. 60, pp. 91–110, 2004.

[11] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection", en, Jun. 2005.

[12] C. Cortes and V. Vapnik, "Support-vector networks", en, *Machine Learning*, vol. 20, no. 3, pp. 273–297, Sep. 1995.

[13] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek, “Image Classification with the Fisher Vector: Theory and Practice”, en, *International Journal of Computer Vision*, vol. 105, no. 3, pp. 222–245, Dec. 2013.

[14] J. Sanchez and F. Perronnin, “High-dimensional signature compression for large-scale image classification”, in *CVPR 2011*, Jun. 2011, pp. 1665–1672.

[15] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition”, in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 770–778. arXiv: 1512.03385.

[16] B. Goodman and S. Flaxman, “European Union regulations on algorithmic decision-making and a "right to explanation"”, *arXiv:1606.08813 [cs, stat]*, Jun. 2016. arXiv: 1606.08813 [cs, stat].

[17] M. T. Ribeiro, S. Singh, and C. Guestrin, “"Why Should I Trust You?": Explaining the Predictions of Any Classifier”, *arXiv:1602.04938 [cs, stat]*, Feb. 2016. arXiv: 1602.04938 [cs, stat].

[18] N. Kingsbury, “Complex wavelets for shift invariant analysis and filtering of signals”, *Applied and Computational Harmonic Analysis*, vol. 10, no. 3, pp. 234–253, May 2001.