Research Journal

Jerome Wynne

University of Bristol

19th June 2017 - Present

Contents

Thursday, 1st June 2017

1 Summary

- Set up this labbook.
- Wrote down what I'd like to achieve during my placement.
- Listed a bunch of classic computer vision papers.

2 Placement objectives

During my 8-week placement I would like to:

- Apply and understand the most useful computer vision algorithms.
- Write either a statistics, machine learning, or computer vision research paper worthy of publication.
- Develop a systematic workflow to problems involving data, and demonstrate this workflow in publicly available scripts or notebooks.

3 Reading a paper critically

- Motivations for the problem posed
- The choices made in finding a solution
- The assumptions behind the solution
- The validity of those assumptions
- Whether assumptions can be removed without invalidating the approach
- What was accomplished
- Future research directions

4 Structuring your time

I think it's realistic to aim for 12 hours of productive work each day. This includes reading, studying, and running my own experiments. To begin with, I think I should spend at least three-quarters of this time learning. Maybe by week four I will shift the ratio.

I will see how it goes, but I suspect I will prefer to work from the University rather than DNV-GL's offices.

To make this placement a success, I will need to be disciplined about how I use my time. I know that if I get up early and immediately go to work, I can easily crack out four hours without breaking a sweat. After this time I can take at least a couple of hours - go to the gym, read, walk, or - of course - eat. After that I'll get back on that horse for another four or five hours, before taking another short break, then do a few more hours.

Having lots of sleep is important for my mental health and emotional wellbeing, so I should aim to get at least eight hours. If I get up at six, this means that I should go to bed at half-nine. I can work from my flat, University buildings, or the DNV-GL offices. It would be a good idea to mix the places I study up - I know this has helped me keep focused in the past. I need to be careful to provide some time for myself too, so that I can recuperate: I think two hours at the end of the day, eight until ten, will be enough.

5 Figuring out what to write about

1. Narrow down your field of study

2. Define what to investigate

3. Establish a thesis or an argument

I am studying computer vision and statistics. This is because I want to build robust algorithms to understand visual information, which in turns makes it easier to automate difficult or tedious tasks, such as watching CCTV cameras or spotting damage to structures in video footage. I am doing this so that we will know more about how patterns in visual information can be found, and so that we can exploit these patterns to automate economically and socially beneficial tasks. I am also interested in how we can represent visual information to make it easier to work with and to understand.

6 Structuring your work

1. Find data that poses an unsolved problem, or find a problem that needs data to be solved.
2. Find the data, or pose the problem.
3. Review the relevant literature.
4. Propose a solution, then run experiments and conduct analysis to test that solution.
5. Write up the results.

7 How to use this book

I should use this book to record what I'm doing, ideas and conversations I have, experiments I run, papers and books to read, things I understand and don't understand - everything related to my research. It only takes a few minutes to put a screenshot in this document - remember this!

As a habit, at the beginning of each day I will write down what I plan to do. At the end of the day I'll review what I've done. I should also review the book each week, to get an idea what I've been up to and where I'm headed.

Papers

Title	Authors	Year	Topic	Read
Theory of Edge Detection	D. Marr, E. Hildreth	1980	Computer vision	No
A Computational Approach to Edge Detection	J. Canny	1986	Computer vision	No
Determining optical flow	B.K.P. Horn, B. Schunk	1981	Computer vision	No
An iterative image registration technique with an application to stereo vision	B. Lucas, T. Kanade	1981	Computer vision	No
Snakes: Active contour models	M. Kass, A. Witkin, D. Terzpoulos	1988	Computer vision	No
Eigenfaces for recognition	M. Turk, A. Pentland	1991	Computer vision	No
Shape and motion from image streams under orthography: a factorization method	C. Tomasi, T. Kanade	1992	Computer vision	No
Texture features for browsing and retrieval of image data	B. Manjunath, W. Ma	1996	Computer vision	No
Conditional density propagation for visual tracking	M. Isard, A. Blake	1998	Computer vision	No
Normalized cuts	J. Shi, J. Malik	2000	Computer vision	No
Non-parametric model for background subtraction	A. Elgammal, D. Harwood, L. Davis	2000	Computer vision	No
Distinctive image features from scale-invariant keypoints	D. Lowe	2004	Computer vision	No

Monday, 19th June 2017

1 Summary

- Arrived at DNV offices @ 10:00.
- H & S induction. Remember to look at the green lights above the doors!
- Met Elizabeth Traiger:
  - Solar panel cracks toy problem.
  - DNV are keen to use Python.
  - Approx. 1500 minutes of turbine data available.
  - We need a method for labeling the data.
- IT was not set up - she will email me tomorrow when it is ready.
- DNV use Microsoft Azure services internally.
- The company is open to using TensorFlow.
- We agreed that weekly meetings would be appropriate. She is away this Friday (in London, speaking about using satellite imagery of windfarms to monitor fish populations (?), in association with the European Space Agency) and next week (site visit?).
- Various ideas came up in our conversation: super-resolution methods, using simulation data, creating synthetic data, thinking about how to segment the images.
- Footage from two turbines is available this week. More footage will be available next week.
- I can VPN into DNV's network using the laptop that they provided me.
- I left the DNV offices after lunch (14:00), then worked on the Design Project briefs until 16:30.
- Liz planned to book a meeting with the inspections team to discuss how they look for damage. Possible questions:
  - What does damage look like?
  - Over what time periods does it develop?

  - What causes the damage?
  - What mistakes do you make when looking for damage?
  - What type of damage is easiest/most difficult to spot?
  - What do you do when you find damage?
  - What problems do you think there are with using drones to identify damage?

In the afternoon I read the introduction to Canny's paper on edge detection. I learnt that there is a tension between edge detection and localization (the distance between the detected edge and the true edge). Apparently the paper determines the optimal operator for various common edges.

I also read the introduction and first chapter of S.J. Prince's book on computer vision. The introduction outlined the book's structure and the first chapter was a quick review of basic probability theory.

- Read Canny paper.
- Read the Chapter 1, 3, and 6 of S.J. Computer Vision. Make notes and complete the associated exercises.
- Apply what was learnt in this chapter using Python.
- Outline the structure of compute vision / machine learning.

2 Prince: Computer Vision (Ch. 1, Introduction)

The book's structure was outlined. It consists of six sections:

1. Probability.
2. Machine learning for machine vision.
3. Graphical models (principled ways to simplify the relationship between image data and estimated properties) for machine vision.
4. Image preprocessing.
5. Geometric machine vision - generating 3D models based on image data and localizing a camera based on its view.
6. Families of machine vision models for common problems.

It took me about ten minutes to read through this first chapter of five pages. Relevant papers are listed at the end of each of the book's sections.

3 Prince: Computer Vision (Ch. 2, Introduction to Probability)

The second chapter took about an hour to read and make notes for. Almost all of it was review material: random variables, distributions, joint distributions, Bayes' rule, conditionality, and expectation. I did learn, however, two useful things:

- A Hinton diagram can be used to visualize probability mass functions. A square is associated with each point; the square's area relative to the total area of all squares corresponds to that value's relative probability. Figure 1 shows a Hinton diagram in two variables.

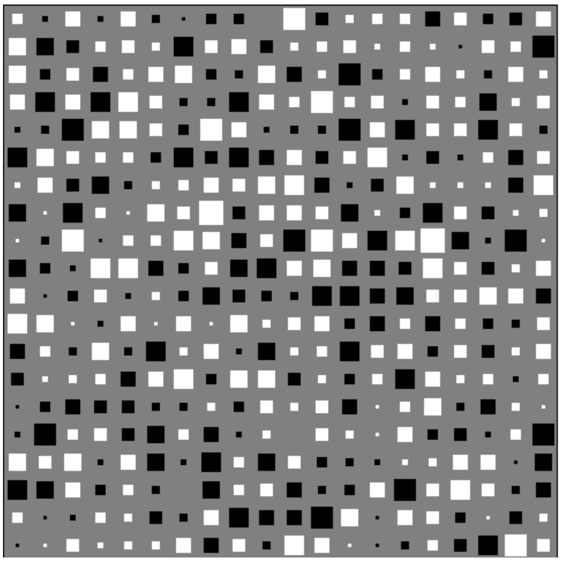

Figure 1: A Hinton diagram can be used to visualize probability mass functions in one or two variables.

- A useful identity: if two random variables are independent, then the product of their expectation is equivalent to the expectation of their product. More generally:

$$E[f(x) \cdot g(y)] = E[f(x)] \cdot E[g(y)] \tag{0.1}$$

Tuesday, 20th June 2017

1 Summary

I woke at six (I'm getting the hang of it). For the first hour of the day I reviewed the project brief MZ had commented on and sent it off to PH. PH mentioned I could use his office for the next two weeks while he was away - hopefully I can pick up the key from him this afternoon.

Today I would like to complete Chapters 3 and 4 of Prince's Computer Vision, do some modeling in Python, and begin the CVX course. I should also get an overview of the main computer vision tools for Python.

I made notes on Chapter 3 and completed five of its exercises, which related to conjugacy, the exponential family, and the modes of the distributions studied. PH said to pick up the key to his office from the porter. ET emailed me to say that IT have set up my account - I agreed to go in tomorrow morning to meet her and pick up my laptop. I should also ask her when we plan to meet the wind turbine inspections team.

I intend to complete Chapter 4 this afternoon. I am now going to read Canny's paper. I tried to read Canny's paper but kept zoning out. I did understand, however, that he formulated an optimization problem by expressing detection performance as signal-to-noise ratio and localization performance as a similar quantity but involving gradients.

I spent fifty minutes reading through Chapter 4. I made notes on maximum likelihood, maximum a posteriori, and bayesian methods for optimizing parameter estimates. I read through an example wherein the mean and variance of a univariate normal distribution were estimated using a normal-scaled inverse gamma prior; as it turns out, the benefits of using the conjugate here are twofold - not only does the posterior have a closed-form solution, but so also does the posterior predictive!

2 Prince: Computer Vision (Ch. 3, Common Probability Distributions)

I started on Chapter 3 of Prince's book yesterday. It covers eight probability distributions that will be useful for machine vision purposes. Four of these are for modelling image data or world properties; the other four are for modelling the parameters of these four.

I made notes, but have yet to complete the exercises. The distribution pairs covered were:

- Bernoulli - Beta
- Categorical - Dirichlet
- Univariate normal - Normal-scaled inverse gamma
- Multivariate normal - Normal inverse Wishart

The four distributions on the left were familiar to me, as was the beta distribution (however it was useful to note that the expectation of Beta$[\alpha, \beta]$ is defined by $\frac{\alpha}{\alpha+\beta}$). The Dirichlet, normal-inverse scaled gamma, and normal inverse wishart distributions were new to me.

The Dirichlet distribution is effectively a generalization of the beta distribution to cover more than one variable. It's used to model the K parameters of a categorical distribution. As such, $\sum_k \lambda_k = 1$ at all points in the Dirichlet distribution. Its hyperparameters $\alpha_1, ..., \alpha_K$ describe the categorical distribution's expecations over each of its variables $E[\lambda_1], ..., E[\lambda_K]$; the magnitude of the α values determines the Dirichlet's dispersion.

The Normal-scaled inverse gamma distribution is used to model uncertainty in the mean and variance of a univariate normal distribution. It has four parameters. In a similar vein, the normal inverse Wishart distribution describes uncertainty in the mean vector and covariance matrix of a multivariate normal distribution.

I'm now going to complete three questions from the exercises, then go to the barbers. When I get back I will complete three more exercises, before continuing to read the Canny paper.

I completed three questions. The first asked for the mode of the beta distribution (found by maximizing the pdf w.r.t. the variable). The second pointed out that all of the distributions in the chapter were members of the exponential family; it then requested the beta pdf in exponential family form. The third related a normal prior and restricted normal likelihood to a normal posterior (i.e. it was a conjugacy problem). These questions together took me about forty-five minutes to complete.

This is not the first time the exponential distribution has been mentioned - why is understanding multiple distributions as variants on one distribution useful?

I completed two more questions: showing that the normal-scaled inverse gamma is the conjugate prior to the univariate normal distribution; showing that the Dirichlet distribution is the conjugate prior to the categorical distribution. They were mostly exercises in algebra, but it did drive home why the Dirichlet and normal-scaled inverse gamma distributions are relevant, and what they look like (particularly w.r.t. the Dirichlet-beta similarities).

3 Prince: Computer Vision (Ch. 4, Fitting Probability Models)

The chapter began with maximum likelihood, maximum a posteriori, and bayesian methods for optimizing parameter estimates. I read through an example in which the mean and variance of a univariate normal distribution were estimated using a normal-scaled inverse gamma prior; as it turns out, the benefits of using the conjugate here are twofold - not only does the posterior have a closed-form solution, but so also does the posterior predictive! I think I would benefit from writing out the Bayesian solution, and possibly running a simulation in Python. Maybe I'll go through the other example before building the simulation.

I made notes on both worked examples. The second example involved inferring posterior and posterior predictive distributions from a categorical likelihood and its conjugate (a Dirichlet prior). As in the case of the univariate normal - normal-scaled inverse gamma problem, bott distributions had neat solutions. The posterior was another Dirichlet distribution (with correspondingly updated parameters $\tilde{\alpha}_j = \alpha_j + N_j$) and the posterior predictive turned out to be a function of beta functions.

I'm going to write two R simulations, one for each example.

I built an R simulation for the first example.

I attempted Q4.6, but fell short (the algebra became untangled). I will try again tomorrow morning; perhaps doing Q4.5 first will make it slightly easier. Once I've done those two questions, I should like to do Q4.7-4.10. Worked 4 - 7:30 on Com. Vis. Chapter 4.

This evening I watched the first lecture from the CVX course. It explained what convex optimisation is, common variants of it (least squares and linear programming), discussed application areas, defined a convex functoin, provided an overview of the course, and gave some history on the topic.

Wednesday, 21st June 2017

1 Summary

Today I plan to:

- Complete the exercises from Ch. 4 of Computer Vision
- Walk through one of the examples available at PyImageSearch
- Read Ch. 5 of Computer Vision and complete the associated exercises
- Pick up Paul Harper's office key
- Meet ET at 10:00 to have IT induction
- Explore solar panel dataset
- Investigate Python tools for tagging images
- Read Chapter 1 of Boyd's Convex Optimisation

I spent the hour and forty-five minutes of the day completing Exercises 4.5 - 4.10 of Computer Vision. I then made notes on Chapter 5, which is focused on the normal distribution. It began by showing the difference between spherical, diagonal, and full covariance matrices - spherical matrices have circular isodensity contours, diagonals have ellipsoidal ones whose axes are aligned with coordinate axes, and full covariance matrices have ellipsoidal contours in any orientation.

As it happens, it's possible to rotate a distribution's coordinate system to permit a full covariance matrix to be reexpressed as a diagonal one. The rotation matrix is deduced using the singular value decomposition (something I should probably know, but don't). I'm about to head off to DNV, but I intend to watch Gilbert Strang's lecture on the singular value decomposition when I get back. I think I could also do with inspecting matrix inverses more closely.