

Search Visualiser, pattern matching, and parallel processing, part 2

By Gordon Rugg

Background: This article unpacks some of the underlying concepts on which Search Visualiser is based.

This article was originally posted on May 8, 2012

The previous article on this topic showed how parallel processing lets you do things faster than step-by-step sequential processing, by doing several things at once. It also showed how using parallel processing to do pattern matching allows you to do things that are difficult or impossible to do via sequential reasoning, such as identifying patterns in where your keywords appear in a document.

One obvious question is: If parallel processing and pattern matching are so great, then why don't we use them for everything?

This article looks at one part of the answer, and at the implications.



(Image copyright Gordon Rugg, 2012)

Good, fast, cheap – choose any two...

The “good, fast, cheap” line is an old joke from engineering and design. It has a habit of being true. It’s true about parallel and sequential processing.

One major reason we don’t use parallel processing for everything is because although it’s good for many purposes, and fast, it’s expensive, either in hardware or in software or in both. That’s the case

whether you're talking about computer hardware and software, or biological hardware and biological information processing. Pattern matching is usually expensive in hardware and or/software, though you can do some types of pattern matching cheaply if time isn't an issue. The expense of parallel processing comes from having to have multiple copies of the hardware that you need to do the task (one copy for each parallel strand) or from needing complex software to handle the process, or some combination of both.

Another major reason that human beings don't use parallel processing for everything is that it has limitations. We use sequential processing a lot for things that are difficult for human parallel processing.

Counting and subitising

We'll start with an example where parallel processing and sequential processing are both reasonable options. How many beads are there in the image below?



(Image copyright Gordon Rugg, 2012)

One way to find the answer is to count the beads one by one, with sequential processing. That will take you five steps.

Another way is to count them by parallel processing. You can just look at them and know that there are five of them. That will take you one step (though it's a step that contains five things happening at the same time).

This method of counting "just by looking" via parallel processing is technically known as subitising. It's a method used not only by adults, but also by children, and by at least some species of animal. Animals and young children can typically handle up to about four items via subitising; adults and older children can typically handle up to about seven items via subitising. People can learn to subitise larger

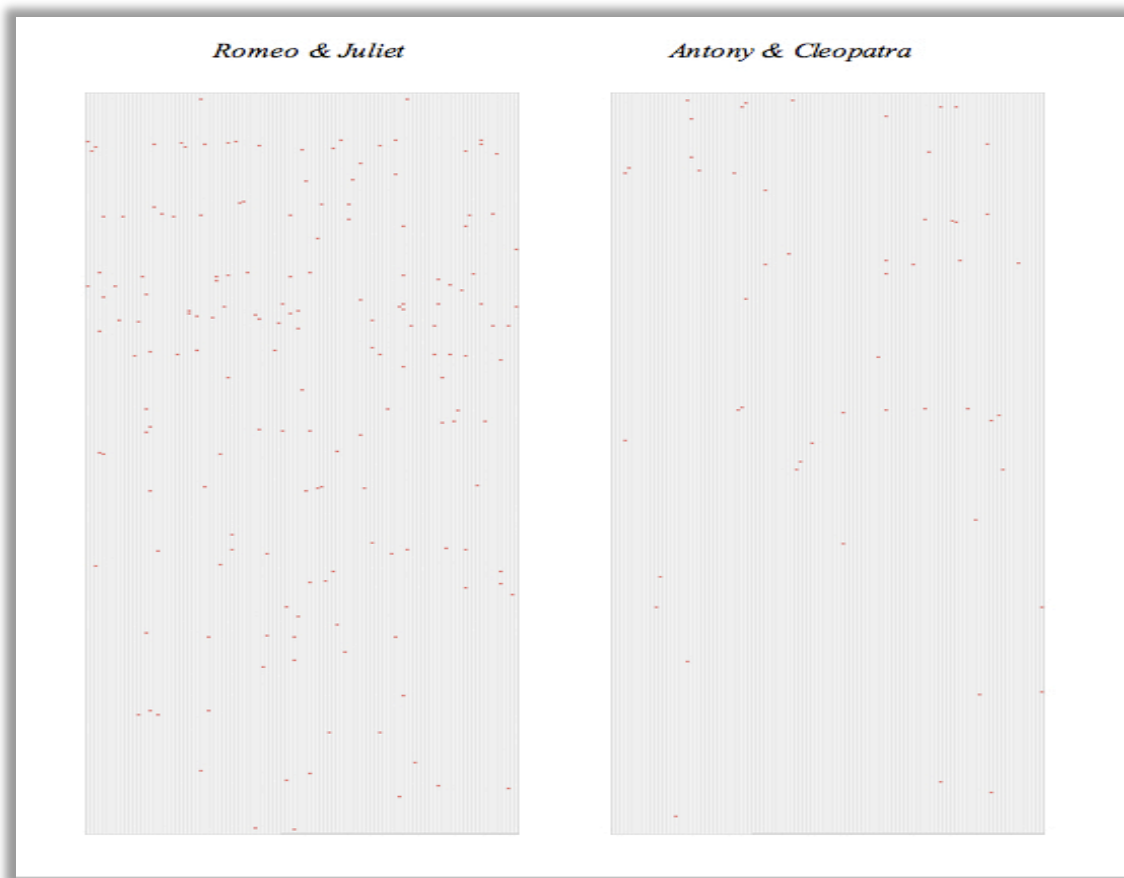
numbers, but it takes practice. Interestingly, there's evidence that parrots, crows and ravens can subitise up to seven items, i.e. the same level of performance as an adult human.

So, there's an upper limit on how high you can count using parallel processing, and it's about seven items. That's a number very familiar to anyone working with human factors and/or psychology. There's a classic paper from 1956 by George Miller entitled "The Magical Number Seven, Plus or Minus Two: Some Limitations on Our Capacity for Processing Information". The title is a pretty good summary of his findings. In all sorts of different areas, you find that the largest number of items that a human can easily handle simultaneously is about seven, plus or minus two. Car registration numbers, phone area codes, individual phone numbers, zip codes and post codes: they're all deliberately kept to a maximum of seven items. It's not a very big number, and that simple fact has a lot of far-reaching implications for how human beings deal with the world.

In principle, there isn't an upper limit on how high you can count using sequential processing. The reality is more complex, and we'll examine that in another article. Counting via sequential processing is slow but steady, and it's easy to automate. So does that mean that we should ignore parallel processing for handling numbers larger than seven plus or minus two?

The answer looks obvious, but it isn't.

Here's an example of how we can handle significantly larger numbers with parallel processing. The two Search Visualiser images below show how often the word "love" is mentioned in two of Shakespeare's great love stories, *Romeo & Juliet* and *Antony & Cleopatra*. Which play contains more mentions of love?



We don't need to count the precise number to answer the question: we can see that one contains far more mentions than the other. As an added bonus, we can see whether there's clustering of where those

mentions occur. In principle, you could describe the clusters statistically, but that wouldn't mean anything to non-statisticians. So, sometimes using parallel processing to handle large numbers is a better choice: it gives you a fast answer, and more information than a number would.

So, summarising the points above:

- For numbers up to about seven, parallel processing and sequential processing are usually both suitable.
- For counting numbers above seven, sequential processing is suitable but parallel processing isn't.
- For comparing how often two things occur, parallel processing and sequential processing can both be suitable, depending on just what you're comparing and how.

Notes

The Search Visualiser is available for online use, free, at:
www.searchvisualiser.com