00:00:00:05 - 00:00:01:12
Measuring the impact of each

00:00:01:12 - 00:00:03:18
optimisation as you go is essential,

00:00:03:18 - 00:00:04:22
otherwise you're guessing.

00:00:04:22 - 00:00:05:17
In practice,

00:00:05:17 - 00:00:07:01
the time.time() approach

00:00:07:01 - 00:00:08:19
I used was straightforward.

00:00:08:19 - 00:00:10:19
Basic timing in the Python script,

00:00:10:19 - 00:00:11:19
using calls,

00:00:11:19 - 00:00:13:16
with CUDA device synchronisation

00:00:13:16 - 00:00:15:05
calls inserted to ensure

00:00:15:05 - 00:00:17:08
the GPU had actually finished its work

00:00:17:08 - 00:00:18:18
before the clock stopped.

00:00:18:18 - 00:00:20:07
Changing one thing at a time

00:00:20:07 - 00:00:21:12
and measuring the before

00:00:21:12 - 00:00:23:08

and after is really all you need

00:00:23:08 - 00:00:23:22
to evaluate

00:00:23:22 - 00:00:26:02
whether a change is helping.

00:00:26:02 - 00:00:28:14
NVIDIA does offer more sophisticated

00:00:28:14 - 00:00:29:19
profiling tools -

00:00:29:19 - 00:00:31:08
Nsight in particular gives

00:00:31:08 - 00:00:32:08
detailed insight

00:00:32:08 - 00:00:33:01
into kernel

00:00:33:01 - 00:00:34:14
execution, memory access

00:00:34:14 - 00:00:36:06
patterns, and occupancy -

00:00:36:06 - 00:00:37:11
and these are worth using

00:00:37:11 - 00:00:38:21
if you have access to them.

00:00:38:21 - 00:00:40:05
But they're not a prerequisite

00:00:40:05 - 00:00:42:00
for making meaningful progress.

00:00:42:00 - 00:00:43:17
And the simple approach got CETRA

00:00:43:17 - 00:00:45:05
to where it needed to be.

00:00:45:05 - 00:00:46:14
The headline result is

00:00:46:14 - 00:00:48:23
the compute speed comparison.

00:00:48:23 - 00:00:49:23
CETRA was benchmarked

00:00:49:23 - 00:00:51:15
against BLS and TLS

00:00:51:15 - 00:00:53:17
across a range of light curve cadences

00:00:53:17 - 00:00:54:24
and durations,

00:00:54:24 - 00:00:55:23
covering the relevant

00:00:55:23 - 00:00:57:24
parameter space for Kepler,

00:00:57:24 - 00:00:59:04
TESS and PLATO.

00:00:59:04 - 00:01:01:10
For PLATO long-cadence targets -

00:01:01:10 - 00:01:02:12
two-year light curves

00:01:02:12 - 00:01:04:04
at 600-second cadence -

00:01:04:04 - 00:01:06:21
CETRA on a single A100 GPU

00:01:06:21 - 00:01:08:08
takes around 3 minutes,

00:01:08:08 - 00:01:09:05
compared to around

00:01:09:05 - 00:01:12:03
2 CPU core-hours for TLS.

00:01:12:03 - 00:01:14:17
That's roughly a 40-fold speedup.

00:01:14:17 - 00:01:16:02
For short-cadence targets

00:01:16:02 - 00:01:18:06
the advantage grows dramatically.

00:01:18:06 - 00:01:19:11
A two-year light curve

00:01:19:11 - 00:01:21:12
at 25-second cadence

00:01:21:12 - 00:01:22:16
takes TLS around

00:01:22:16 - 00:01:24:13
100 CPU core-hours,

00:01:24:13 - 00:01:26:05
roughly 4 days of compute

00:01:26:05 - 00:01:28:04
time on a single core.

00:01:28:04 - 00:01:29:21
CETRA handles it in a similar time

00:01:29:21 - 00:01:31:08
to the 600 second case,

00:01:31:08 - 00:01:32:24

since the periodic search -

00:01:32:24 - 00:01:33:18
which dominates

00:01:33:18 - 00:01:34:17
the run time -

00:01:34:17 - 00:01:36:09
depends on the epoch baseline

00:01:36:09 - 00:01:38:05
rather than the cadence.

00:01:38:05 - 00:01:42:00
On a workstation-grade NVIDIA RTX A5000,

00:01:42:00 - 00:01:43:10
CETRA is only around

00:01:43:10 - 00:01:45:16
one fifth slower than on the A100,

00:01:45:16 - 00:01:47:07
despite the A100 costing

00:01:47:07 - 00:01:49:11
roughly four times as much.

00:01:49:11 - 00:01:51:03
That's still a very large improvement

00:01:51:03 - 00:01:53:03
over CPU-based alternatives,

00:01:53:03 - 00:01:54:02
and it means CETRA

00:01:54:02 - 00:01:55:04
is practical to run

00:01:55:04 - 00:01:56:19
and develop on a workstation,

00:01:56:19 - 00:01:58:09
not just on a cluster.

00:01:58:09 - 00:01:59:14
To put the compute speed

00:01:59:14 - 00:02:00:23
in practical terms:

00:02:00:23 - 00:02:03:16
for a typical PLATO long-cadence target,

00:02:03:16 - 00:02:04:20
TLS takes around

00:02:04:20 - 00:02:06:15
2 CPU core-hours,

00:02:06:15 - 00:02:08:14
rising to around 10 for

00:02:08:14 - 00:02:10:06
a four-year light curve.

00:02:10:06 - 00:02:11:22
CETRA reduces this to minutes

00:02:11:22 - 00:02:13:22
per target on a single GPU.

00:02:13:22 - 00:02:15:11
That shift - from hours

00:02:15:11 - 00:02:16:07
to minutes -

00:02:16:07 - 00:02:17:18
transforms what's feasible

00:02:17:18 - 00:02:19:06
at survey scale.

00:02:19:06 - 00:02:20:19
A large GPU cluster can

00:02:20:19 - 00:02:21:09
of course

00:02:21:09 - 00:02:22:10
process enormous

00:02:22:10 - 00:02:24:09
numbers of targets very rapidly,

00:02:24:09 - 00:02:26:01
but the efficiency gains mean

00:02:26:01 - 00:02:27:20
that even a single workstation-grade

00:02:27:20 - 00:02:30:09
GPU is enough to make large-scale

00:02:30:09 - 00:02:32:22
transit searches practical.

00:02:32:22 - 00:02:34:19
Research groups without dedicated

00:02:34:19 - 00:02:37:02
HPC access can now run searches

00:02:37:02 - 00:02:37:22
that simply wouldn't

00:02:37:22 - 00:02:39:07
have been feasible before.

00:02:39:07 - 00:02:40:17
It's worth understanding why

00:02:40:17 - 00:02:42:02
the speedup is so dramatic

00:02:42:02 - 00:02:44:16

for short-cadence data specifically,

00:02:44:16 - 00:02:45:14
because it illustrates

00:02:45:14 - 00:02:46:09
something important

00:02:46:09 - 00:02:48:06
about how CETRA is structured.

00:02:48:06 - 00:02:49:11
In TLS,

00:02:49:11 - 00:02:51:15
the total computational cost scales

00:02:51:15 - 00:02:53:07
with the number of data points.

00:02:53:07 - 00:02:54:03
More points

00:02:54:03 - 00:02:55:23
means more work at every step,

00:02:55:23 - 00:02:57:06
including the phase-folding

00:02:57:06 - 00:02:59:03
and sorting operations.

00:02:59:03 - 00:02:59:19
A two-year

00:02:59:19 - 00:03:00:09
light curve at

00:03:00:09 - 00:03:01:07
25 second

00:03:01:07 - 00:03:02:14
cadence has around

00:03:02:14 - 00:03:04:18
2.5 million data points, and TLS

00:03:04:18 - 00:03:06:11
has to sort and process

00:03:06:11 - 00:03:09:00
all of them for every trial period.

00:03:09:00 - 00:03:11:11
In CETRA, the linear search cost does

00:03:11:11 - 00:03:13:05
scale with the number of data points,

00:03:13:05 - 00:03:14:22
but the linear search is fast

00:03:14:22 - 00:03:17:07
and it only needs to be run once.

00:03:17:07 - 00:03:18:10
The periodic search,

00:03:18:10 - 00:03:20:10
which is the expensive part,

00:03:20:10 - 00:03:22:00
operates not on the raw light curve,

00:03:22:00 - 00:03:22:24
but on the compact

00:03:22:24 - 00:03:25:12
output of the linear search -

00:03:25:12 - 00:03:28:01
those 2D arrays of depth, depth variance

00:03:28:01 - 00:03:29:15
and likelihood ratio,

00:03:29:15 - 00:03:31:02
whose size is determined

00:03:31:02 - 00:03:32:07
by the number of trial

00:03:32:07 - 00:03:34:05
transit times and durations,

00:03:34:05 - 00:03:36:20
not by the cadence of the original data.

00:03:36:20 - 00:03:38:12
So increasing the cadence

00:03:38:12 - 00:03:39:15
makes the linear search

00:03:39:15 - 00:03:40:21
a little more expensive,

00:03:40:21 - 00:03:42:16
but leaves the periodic search

00:03:42:16 - 00:03:44:16
entirely unaffected.

00:03:44:16 - 00:03:47:11
This separation of the cadence-dependent

00:03:47:11 - 00:03:48:14
and cadence-independent

00:03:48:14 - 00:03:49:22
parts of the computation

00:03:49:22 - 00:03:50:17
is one of the most

00:03:50:17 - 00:03:53:12
consequential design decisions in CETRA.

00:03:53:12 - 00:03:55:08

Improved sensitivity was a secondary

00:03:55:08 - 00:03:56:22
goal of CETRA’s design -

00:03:56:22 - 00:03:58:14
the primary goal was speed -

00:03:58:14 - 00:03:59:17
but it's a welcome outcome

00:03:59:17 - 00:04:01:10
of getting the algorithm right.

00:04:01:10 - 00:04:03:00
To evaluate sensitivity,

00:04:03:00 - 00:04:03:21
synthetic light

00:04:03:21 - 00:04:04:23
curves were generated

00:04:04:23 - 00:04:06:22
at three different white noise levels,

00:04:06:22 - 00:04:08:19
each containing an injected transit

00:04:08:19 - 00:04:10:05
signal with randomly drawn

00:04:10:05 - 00:04:12:13
period, duration and depth.

00:04:12:13 - 00:04:13:08
Both CETRA

00:04:13:08 - 00:04:14:13
and TLS were run on

00:04:14:13 - 00:04:15:24
identical light curves,

00:04:15:24 - 00:04:17:21
and the fraction of correctly recovered

00:04:17:21 - 00:04:19:14
periods was compared.

00:04:19:14 - 00:04:20:21
At low noise levels,

00:04:20:21 - 00:04:23:09
the two algorithms performed similarly -

00:04:23:09 - 00:04:24:13
when the signal is strong enough

00:04:24:13 - 00:04:26:24
both find it reliably.

00:04:26:24 - 00:04:28:06
The differences emerge

00:04:28:06 - 00:04:29:13
at higher noise levels

00:04:29:13 - 00:04:31:22
and for harder-to-detect signals.

00:04:31:22 - 00:04:33:12
In the hardest cases tested

00:04:33:12 - 00:04:34:08
CETRA recovered

00:04:34:08 - 00:04:37:14
21% more signals overall than TLS.

00:04:37:14 - 00:04:38:20
For the subset of signals

00:04:38:20 - 00:04:39:19
corresponding roughly

00:04:39:19 - 00:04:41:06
to Earth-Sun analogues,

00:04:41:06 - 00:04:43:08
the improvement is larger still.

00:04:43:08 - 00:04:44:19
The sensitivity improvement

00:04:44:19 - 00:04:46:03
is a direct consequence

00:04:46:03 - 00:04:47:10
of the design decisions

00:04:47:10 - 00:04:49:03
already described.

00:04:49:03 - 00:04:51:12
Resampling handles data gaps formally,

00:04:51:12 - 00:04:53:09
so a gap that falls within a transit

00:04:53:09 - 00:04:55:06
window doesn't bias the result -

00:04:55:06 - 00:04:56:08
it simply reduces

00:04:56:08 - 00:04:57:10
the weight of that transit

00:04:57:10 - 00:04:59:03
in the joint likelihood.

00:04:59:03 - 00:05:00:03
The fixed transit

00:05:00:03 - 00:05:01:17
reference time step ensures

00:05:01:17 - 00:05:03:06

uniform sensitivity across

00:05:03:06 - 00:05:05:15
all period and duration combinations,

00:05:05:15 - 00:05:06:18
and the inverse-variance

00:05:06:18 - 00:05:07:19
weighting of the depth

00:05:07:19 - 00:05:09:01
estimate makes proper use

00:05:09:01 - 00:05:10:11
of the measurement uncertainties

00:05:10:11 - 00:05:11:13
rather than treating

00:05:11:13 - 00:05:13:03
all data points equally.

00:05:13:03 - 00:05:15:00
None of the improvements in sensitivity

00:05:15:00 - 00:05:16:20
came from the optimisations -

00:05:16:20 - 00:05:18:06
they were the natural consequence

00:05:18:06 - 00:05:20:07
of building the algorithm correctly.

00:05:20:07 - 00:05:21:18
The sensitivity improvements

00:05:21:18 - 00:05:22:23
came along for the ride.

00:05:22:23 - 00:05:24:24
No demonstration of a transit detection

00:05:24:24 - 00:05:25:23
algorithm is complete

00:05:25:23 - 00:05:28:00
without a test on real data.

00:05:28:00 - 00:05:29:08
We applied CETRA to the

00:05:29:08 - 00:05:34:14
HD 101581 system, also known as TOI-6276 -

00:05:34:14 - 00:05:35:19
a K-type star

00:05:35:19 - 00:05:38:10
observed by TESS at 20-second cadence,

00:05:38:10 - 00:05:39:06
known to host

00:05:39:06 - 00:05:40:24
at least two Earth-sized planets

00:05:40:24 - 00:05:42:11
and a third candidate.

00:05:42:11 - 00:05:44:00
The two confirmed planets,

00:05:44:00 - 00:05:49:06
TOI-6276.01 and TOI-6276.02

00:05:49:06 - 00:05:49:20
were recovered

00:05:49:20 - 00:05:51:05
with signal-to-noise ratios

00:05:51:05 - 00:05:53:19
of around 20 and 23 respectively,

00:05:53:19 - 00:05:54:20
a modest improvement

00:05:54:20 - 00:05:55:22
over the TLS results

00:05:55:22 - 00:05:58:04
in the existing literature.

00:05:58:04 - 00:05:59:08
The more interesting result

00:05:59:08 - 00:06:01:02
concerns the third candidate,

00:06:01:02 - 00:06:03:23
TOI-6276.03,

00:06:03:23 - 00:06:05:11
which had previously been detected

00:06:05:11 - 00:06:08:09
at a signal-to-noise ratio of 7.9,

00:06:08:09 - 00:06:09:19
too low to meet the threshold

00:06:09:19 - 00:06:11:04
for statistical validation

00:06:11:04 - 00:06:12:04
as a planet.

00:06:12:04 - 00:06:16:11
CETRA recovers TOI-6276.03

00:06:16:11 - 00:06:18:16
at a signal-to-noise ratio of 16.

00:06:18:16 - 00:06:19:23
That's a factor of two

00:06:19:23 - 00:06:21:19

improvement in significance.

00:06:21:19 - 00:06:23:11
The authors who originally identified

00:06:23:11 - 00:06:25:06
this candidate noticed that its false

00:06:25:06 - 00:06:26:10
positive probability

00:06:26:10 - 00:06:27:15
was already low enough

00:06:27:15 - 00:06:30:04
to satisfy the validation criteria

00:06:30:04 - 00:06:33:00
if the SNR were higher -

00:06:33:00 - 00:06:34:00
the low SNR

00:06:34:00 - 00:06:36:00
was the only obstacle.

00:06:36:00 - 00:06:37:05
At SNR 16,

00:06:37:05 - 00:06:39:03
it may now be possible to reconsider

00:06:39:03 - 00:06:40:11
the statistical validity

00:06:40:11 - 00:06:43:02
of this candidate as a confirmed planet.

00:06:43:02 - 00:06:44:01
The full analysis -

00:06:44:01 - 00:06:45:12
including 3 successive

00:06:45:12 - 00:06:46:13
detection runs,

00:06:46:13 - 00:06:48:17
each masking the most significant signal

00:06:48:17 - 00:06:50:05
before re-running -

00:06:50:05 - 00:06:54:18
took 2.6 seconds on an RTX A5000 GPU.

00:06:54:18 - 00:06:56:11
The equivalent TLS analysis

00:06:56:11 - 00:06:58:21
parallelised across 6 CPU cores,

00:06:58:21 - 00:07:00:09
took around 4 minutes.

00:07:00:09 - 00:07:02:10
That's a factor of 90 reduction

00:07:02:10 - 00:07:03:20
in wall-clock time.

00:07:03:20 - 00:07:05:10
CETRA is open source

00:07:05:10 - 00:07:06:19
published on GitHub

00:07:06:19 - 00:07:09:12
and available to install via PyPI.

00:07:09:12 - 00:07:12:01
It requires an NVIDIA GPU.

00:07:12:01 - 00:07:13:08
Any reasonably modern

00:07:13:08 - 00:07:14:17
NVIDIA card should work,

00:07:14:17 - 00:07:15:17
though the more powerful

00:07:15:17 - 00:07:17:19
the card, the faster it will run.

00:07:17:19 - 00:07:20:13
The GitHub repository includes example

00:07:20:13 - 00:07:21:21
Jupyter notebooks,

00:07:21:21 - 00:07:25:14
including the full HD 101581 analysis.

00:07:25:14 - 00:07:27:00
These are the best starting point

00:07:27:00 - 00:07:28:14
for new users.

00:07:28:14 - 00:07:30:06
The code is written to be run

00:07:30:06 - 00:07:31:17
on detrended light curves -

00:07:31:17 - 00:07:32:22
CETRA does not perform

00:07:32:22 - 00:07:34:04
detrending itself -

00:07:34:04 - 00:07:35:14
and uses sensible defaults

00:07:35:14 - 00:07:37:02
for all search parameters,

00:07:37:02 - 00:07:38:22

so getting a first result requires

00:07:38:22 - 00:07:40:08
minimal configuration.

00:07:40:08 - 00:07:42:12
Documentation is available

00:07:42:12 - 00:07:44:00
at cetra.readthedocs.io.

00:07:44:00 - 00:07:46:01
For those who want to dig into the code,

00:07:46:01 - 00:07:47:11
the CUDA kernel source

00:07:47:11 - 00:07:48:15
is in the repository

00:07:48:15 - 00:07:50:04
and is written to be readable.

00:07:50:04 - 00:07:51:04
The design decisions

00:07:51:04 - 00:07:51:19
described in

00:07:51:19 - 00:07:53:10
this series of videos are reflected

00:07:53:10 - 00:07:55:08
directly in the code structure,

00:07:55:08 - 00:07:56:12
so it should be possible

00:07:56:12 - 00:07:58:05
to trace the path from the ideas

00:07:58:05 - 00:08:01:01
described here to their implementation.