Optimising a pitch tracker

- Downsampling
- **Sliding windows**
- Neighbourhood search

For a given lag, say 100, when we move along 1 sample then 98 of the mul-adds are the same as for the previous position

Because we only care about the sum (for E and for H) we can remove the oldest value and add one new value

Reduces $O(N^2)$ to $O(N)$

Known as the “sliding window” algorithm

Optimising a pitch tracker

- Downsampling
- Sliding windows
- **Neighbourhood search**

After getting a pitch at 5kHz, we can examine the neighbourhood (+/- 8 lags, called N) at the full resolution and refine the estimate

This requires recalculating H at full resolution which is expensive.

Only performed every 40 samples at full resolution

If the peak chosen is in the top half of the neighbourhood, we shift the neighbourhood “up” by one lag, which helps to track changes in frequency over time. And vice-versa.