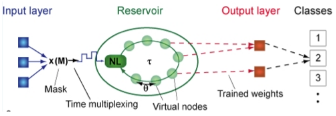
Recognising a friend’s face in a crowd of people is a complex task for the brain, yet it might take only a fraction of a second. It is one of the biggest unresolved questions in brain research, how the electrical discharges of billions of neurons are organised to deliver correct answers in such a short time. However, in recent years a paradigm has been developed in neuroscience that might lead to an answer.

Neuroscientists exploit an analogy between the human brain’s response to external stimuli and the reaction of a liquid to external perturbations, such as a pebble thrown into water. From the waves generated by the impact, one can conclude where and when the pebble hit the surface. Similarly, it could be possible to draw information about external stimuli from the transient responses of neural networks. These networks into which stimuli or inputs are fed are referred to as ‘reservoirs’ and the technique is called “reservoir computing”.

So, how does reservoir computing work? Like a pebble rippling the water surface, external stimuli or ’inputs’ remain detectable in the reservoir for a certain time. This memory of the input, in combination with the emerging response of the reservoir, transforms the input into a large number of dynamical states of the reservoir. The trick of reservoir computing is that the reservoir’s responses to different inputs are easier to classify than the inputs.

The PHOCUS consortium has identified a novel approach to realise a reservoir with one or few photonic components by utilising delay coupled semiconductor lasers. The realisation of reservoir computing using photonic systems offers great opportunities. Photonic systems allow for extremely fast processing and are compatible with telecommunication technology.

Ultimately, PHOCUS aims at photonic implementations of reservoir computing operating at high data rates. This would offer alternatives to supercomputers and computer clusters for specific tasks requiring reduced size and low power consumption.



Scheme of Reservoir Computing utilising a nonlinear node with delayed feedback. A reservoir is obtained by dividing the delay loop into N intervals and using time multiplexing. The input states are sampled and held for a duration τ ,where τ is the delay in the feedback loop. For any time t0 the input state is multiplied with a mask, resulting in a temporal input stream J(t) which is added to the delayed state of the reservoir and then fed into the nonlinear node. The output nodes are linear weighted sums of the tapped states in the delay line.