Ray-based classification framework for high-dimensional data

Justyna P. Zwolak[1], Sandesh S. Kalantre[2], Thomas McJunkin[3], Brian J. Weber[4], and Jacob M. Taylor[1,2]

[1] National Institute of Standards and Technology, Gaithersburg, MD 20899, USA [2] University of Maryland, College Park, MD 20742, USA
[3] University of Wisconsin-Madison, Madison, WI 53706, USA [4] ShanghaiTech University, Shanghai 201210, China

NIST — Joint Center for Quantum Information and Computer Science — Wisconsin, University of Wisconsin–Madison

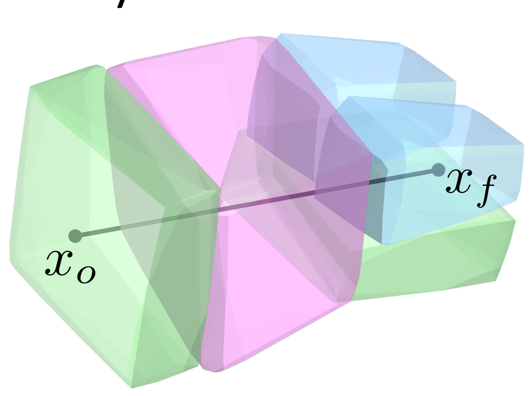
Motivation

- While classification of arbitrary structures in high dimensions may require complete quantitative information, for simple geometrical structures, low-dimensional qualitative information about the boundaries defining the structures can suffice.
- We propose a deep neural network (DNN) classification framework that utilizes a minimal collection of one-dimensional representations, called *rays*, to construct the *fingerprint* of the structure(s) based on substantially reduced information.

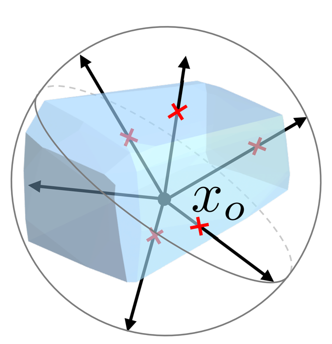
Ray-based framework

- Consider Euclidean space $\mathbb{R}^N$ with its conventional 2-norm distance function d and a polytope function $p\colon \mathbb{R}^N \to \{0,1\}$. The set of points where $p(x)=1$ constitutes the boundary of a collection of polytopes.

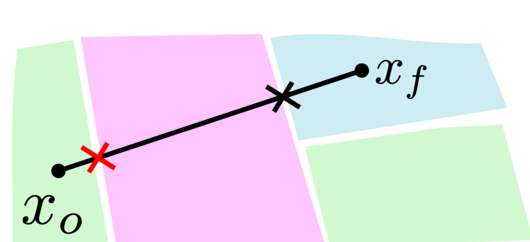
Given $x_o, x_f \in \mathbb{R}^N$, a set of points $\mathfrak{R}_{x_o,x_f} := \{(1-t)x_o + t x_f, t \in [0,1]\}$ defines a __ray__ from x_o to x_f.



- Consider a collection of M rays of a fixed length r, $\mathcal{R}_M := \{\mathfrak{R}_{x_o,x_m}, m = 1,\dots,M\}$ centered at x_o.



Given a point $x \in \mathfrak{R}_{x_o,x_f}$ and a polytope p, x is a __feature__ if $p(x)=1$.



Features along a given ray define its __feature set__, $F_{x_o,x_f} := \{x \in \mathfrak{R}_{x_o,x_f} \mid p(x)=1\}$ with a natural order given by the 2-norm distance function $d\colon x_o \times F_{x_o,x_f} \to \mathbb{R}^+$.

- Consider a decreasing weight function $\gamma\colon \mathbb{R}^+ \to [0,1]$, a __weight set__ $\Gamma_{x_o,x_f} = \{\gamma(d(x,x_f)) \mid x \in F_{x_o,x_f}\}$ corresponding to the feature set, and a point $x \in \Gamma_{x_o,x_f}$ with highest (i.e., critical) weight W_{x_o,x_f}.

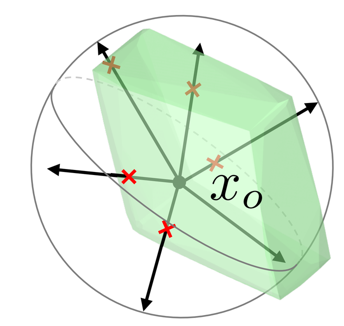
Let $x_o \in \mathbb{R}^N$ be a point from which a collection of rays $\mathcal{R}_M$ emanate. The __point fingerprint__ of x_o is the M-dimensional vector consisting of the rays' critical weights:

$$\mathcal{F}_{x_o} := \left(W_{x_o,x_f^1}, \dots, W_{x_o,x_f^M}\right),$$

Where $W_{x_o,x_f^k} = 0$ if $\Gamma_{x_o,x_f^k} = \emptyset$.

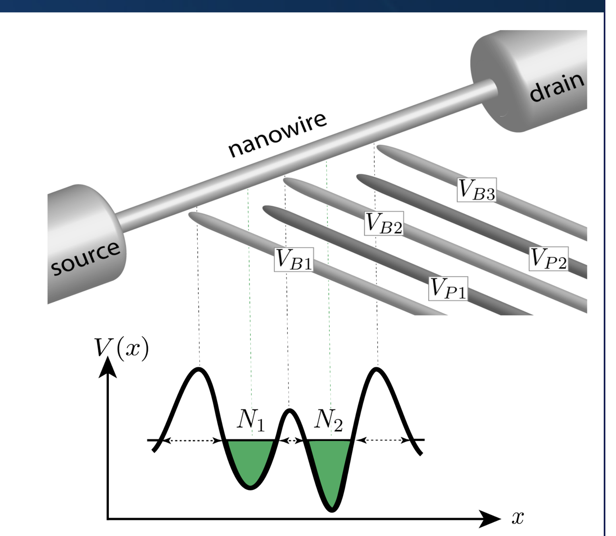
Problem formulation

Given a set of bounded and unbounded convex polytopes filling an N-dimensional space and belonging to C distinct classes ($C \in \mathbb{N}$), and a point $x_o \in \mathbb{R}^N$, determine to which of the classes the polytope enclosing x_o belongs.

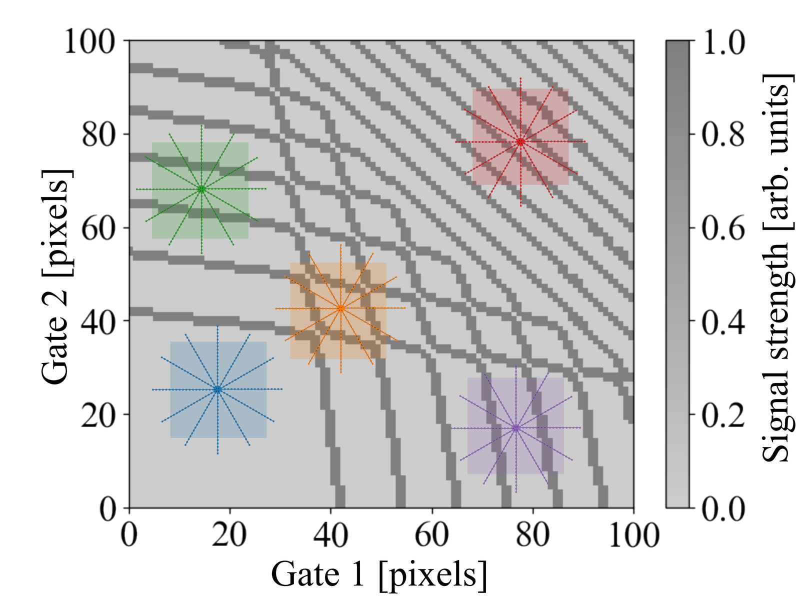


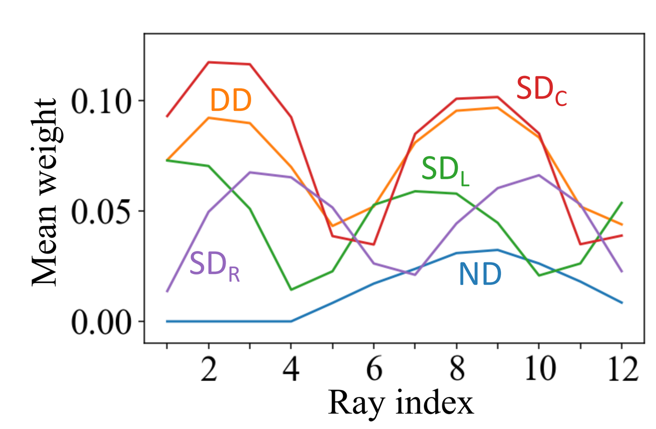
Experiments: 2D quantum dots

- Electrons in quantum dot can be used to define *qubits*.[1]
- Electrons are confined via electrostatic potential created by gates.
- Voltage on each gate needs to be set to bring the device into a desirable regime of operation.[2]

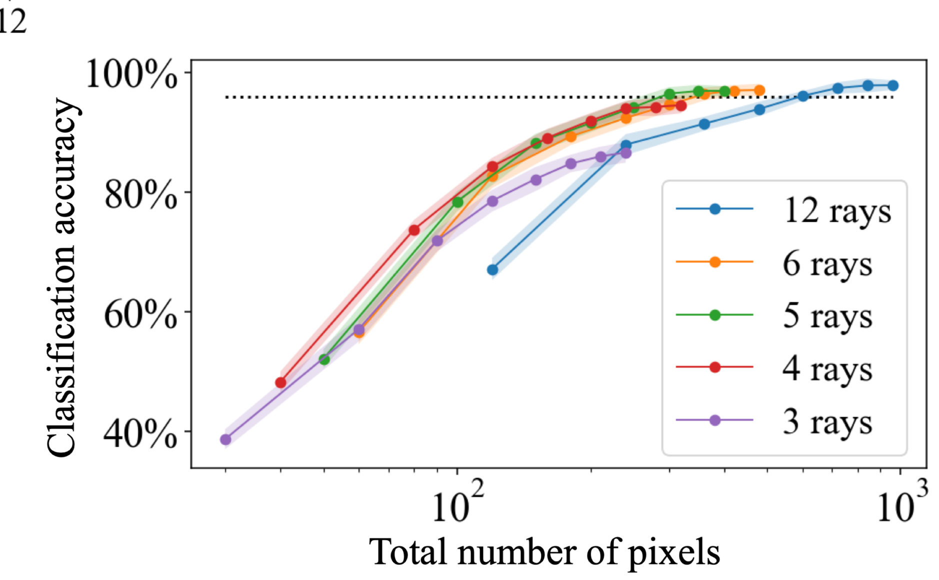


A sample 2D map generated with the quantum dot simulator[3] showing the different bounded and unbounded polytopes in $\mathbb{R}^2$ with 12 evenly distributed rays overlaid on 2D scans like the ones used in Ref. [3].

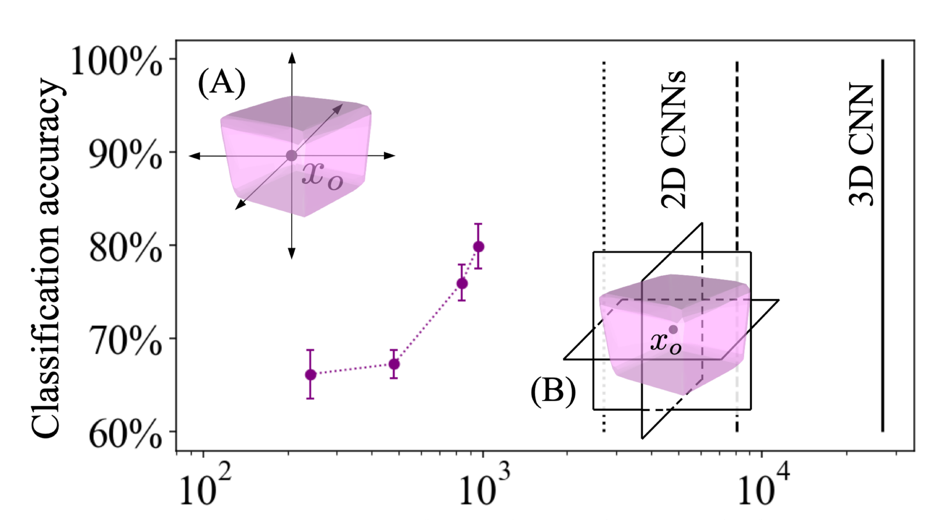




The average trends for fingerprints.



Classifier performance for varying numbers of rays as a function of the total number of pixels measured and averaged over 50 training runs.



Classifier performance for varying numbers of rays as a function of the total number of voxels measured and averaged over 10 training runs.

Ray-based fingerprinting algorithm

Algorithm 1 Ray-based fingerprinting algorithm

Step 1. *Find M-projection centered at x_o given r.*

```
Input: x_o, r, a set P of M points on the (N − 1)-sphere
m ← 1; R_M ← empty list
for m = 1 to M do
    Find m-th ray ℜ_{x_o,x_f^m} and append it to the list R_M.
end for
Return: List of M rays R_M.
```

Step 2. *Fingerprint $x_o \in \mathbb{R}^N$ using rays in $\mathcal{R}_M$ from Step 1.*

```
Input: R_M, γ : ℝ⁺ → [0, 1]
m ← 1; F_{x_o} ← empty list
for m = 1 to M do
    Find the feature set F_{x_o,x_f^m}.
    if F_{x_o,x_f^m} ≠ ∅ then
        Identify the critical feature x_i^m, find W_{x_o,x_f^m} and append
        it to the list F_{x_o}.
    else
        Append 0 to the list F_{x_o}.
    end if
end for
Return: The point fingerprint vector F_{x_o}.
```

Summary

- We have defined a framework to generate a low-dimensional representation of geometrical shapes in a high-dimensional space.
- We have empirically shown that the proposed framework is an effective solution for cutting down the measurement cost while preserving high-accuracy of classification on the quantum dot dataset. The ray-based classifier lead to results on par with the CNN based classifier (96.4 ± 0.4) % while reducing the data requirement by 60 %. This promises significant improvements if implemented in a scheme to tune double quantum dots in experiments.
- Out preliminary analysis suggests that the reduction in data requirements for 3D data is even more significant.

References

[1] D. Loss and D.P. DiVincenzo. Quantum computation with quantum dots. *Phys. Rev. A*, **57**, 120 (1998).
[2] D. M. Zajac, T.M. Hazard, X. Mi *et al.* Scalable Gate Architecture for a One-Dimensional Array of Semiconductor Spin Qubits. *Phys. Rev. Appl.* **6**, 054013 (2016).
[3] J.P. Zwolak, S.S. Kalantre, X. Wu *et al.* QFlow lite dataset: A machine-learning approach to the charge states in quantum dot experiments. *PLOS ONE* **13**(10): e0205844 (2018).
[4] S.S. Kalantre, J.P. Zwolak, S. Ragole *et al.* Machine learning techniques for state recognition and auto-tuning in quantum dots. *npj Quantum Inf.* **5**, 6 (2019).