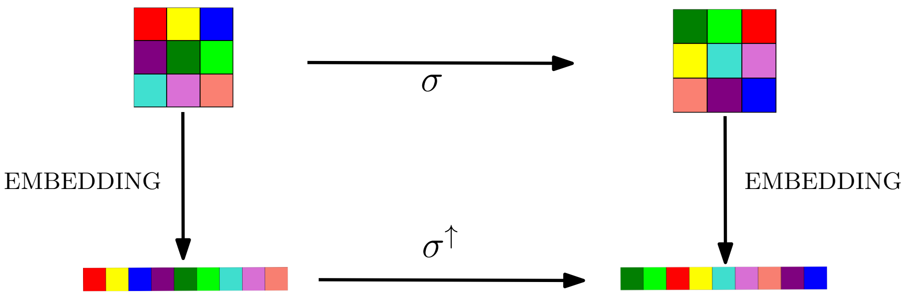
Detection of compact Lie group representations in point clouds and image data

Henrique Ennes and Raphaël Tinarrage — FGV (Rio de Janeiro, Brazil)

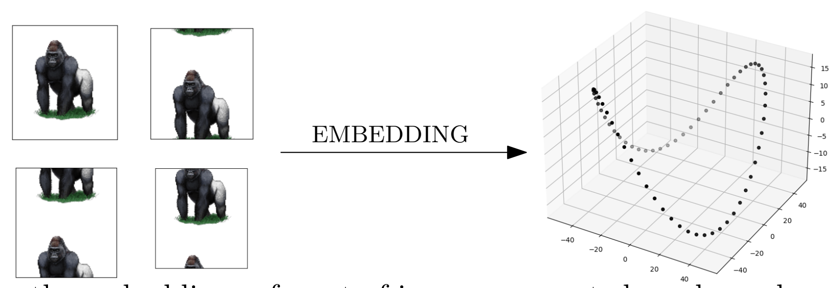
Transformations by permuting pixels

We consider two $m \times m$-pixeled images, f and f', such that there exists a **permutation of pixels** σ for which $f' = \sigma \cdot f$.

In the embedding space $\mathbb{R}^{m \times m}$, this implies the existence of a **permutation matrix** $\sigma^{\uparrow}$ such that the lifted images are related by $f'^{\uparrow} = \sigma^{\uparrow} \cdot f^{\uparrow}$.



Although discrete, the embedding of a set of images $\{f_i\}$ generated by the application of a group of permutations Σ lies close to a **smooth geometric structure** in $\mathbb{R}^{m \times m}$, whose nature may be useful in several computer vision problems.



Lemma: the embeddings of a set of images generated as above described lie on the orbit of some **compact Lie group representation**.

Theory of compact Lie groups

A group of $n \times n$ matrices, $G \subseteq GL(n, \mathbb{F})$, is a **compact Lie group** if, besides having its usual matrix product $\cdot$ and inversion smooth, is endowed with a compact manifold structure in $\mathbb{F}^{n \times n}$ ($\mathbb{F} = \mathbb{R}$ or $\mathbb{F} = \mathbb{C}$).

Lie group	Symbol	Definition	dim	Lie algebra
orthogonal group	$O(n)$	$A^T = A^{-1}$	$\frac{n(n-1)}{2}$	$\mathfrak{o}(n)$
special orthogonal group	$SO(n)$	$A^T = A^{-1}$, det $A = 1$	$\frac{n(n-1)}{2}$	$\mathfrak{so}(n)$
torus group	T^n	$SO(2)^n$	n	$\mathfrak{t}(n)$
unitary group	$U(n)$	$A^\dagger = A^{-1}$	n^2	$\mathfrak{u}(n)$
special unitary group	$SU(n)$	$A^\dagger = A^{-1}$, det $A = 1$	$n(n-1)$	$\mathfrak{su}(n)$

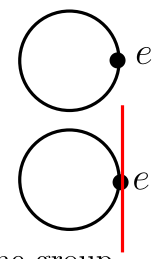
The tangent space at the identity of a Lie group G is called its **Lie algebra**, denoted by $\mathfrak{g}$, and forms a well-understood structure closed under the usual matrix commutation $[\cdot,\cdot]$.

Example:

$$SO(2) \quad \begin{pmatrix} \cos t & -\sin t \\ \sin t & \cos t \end{pmatrix}$$

GEOMETRIC REALIZATION

$$\mathfrak{so}(2) \quad \begin{pmatrix} 0 & -t \\ t & 0 \end{pmatrix} \qquad \approx$$



The Lie algebra allows for a "linear" simplification of the group.
Example: while the brackets of $\mathfrak{t}^3$ are trivial, for $\mathfrak{so}(3)$, they are isomorphic to the usual cross-product in $\mathbb{R}^3$.

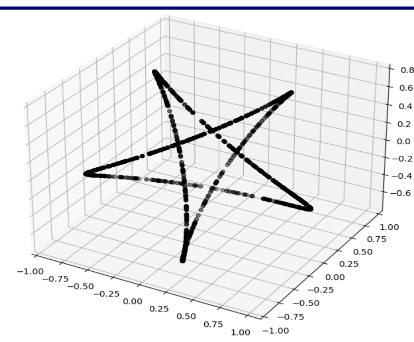
A **representation** of a compact Lie group G is a smooth homomorphism $\phi : G \to GL(n, \mathbb{R})$, making a **copy** of G on $GL(n, \mathbb{R})$, also making a copy of the Lie algebra $\mathfrak{g}$ on $M_{n \times n}$ by the **derived representation** $d\phi$.
Given a compact Lie group G:

- the set of all its representations are known and discrete, up to a change of basis.
- all its representations in $GL(n, \mathbb{R})$ can be transformed, by a change of basis, to be **orthogonal** (i.e., a subgroup of $O(n)$).

Example:

$$SO(2) = \begin{pmatrix} \cos t & -\sin t \\ \sin t & \cos t \end{pmatrix} \overset{\phi}{\Longrightarrow} \begin{pmatrix} \cos 2t & -\sin 2t & 0 & 0 \\ \sin 2t & \cos 2t & 0 & 0 \\ 0 & 0 & \cos 3t & -\sin 3t \\ 0 & 0 & \sin 3t & \cos 3t \end{pmatrix}$$

$$\mathfrak{so}(2) = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \overset{d\phi}{\Longrightarrow} \begin{pmatrix} 0 & -2 & 0 & 0 \\ 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & -3 \\ 0 & 0 & 3 & 0 \end{pmatrix}$$



The algorithm

`Input`: a finite sample of points $\{x_i\}_{i=1}^N$ close or included in an **orbit** $\mathcal{O} = \phi(G) \cdot x_1$ of a compact Lie group G with a representation ϕ on the embedding space $\mathbb{R}^n$.
`Output`: an estimation of the orbit, $\hat{\mathcal{O}}$ and the derived representation $(d\hat{\phi}, [\cdot,\cdot])$.

`Step 1`: orthogonalize the representation through $\mathcal{O} \leftarrow M\mathcal{O}$ for $M = \left(\frac{1}{N}\sum_{i=1}^N x_i x_i^T\right)^{-1/2}$

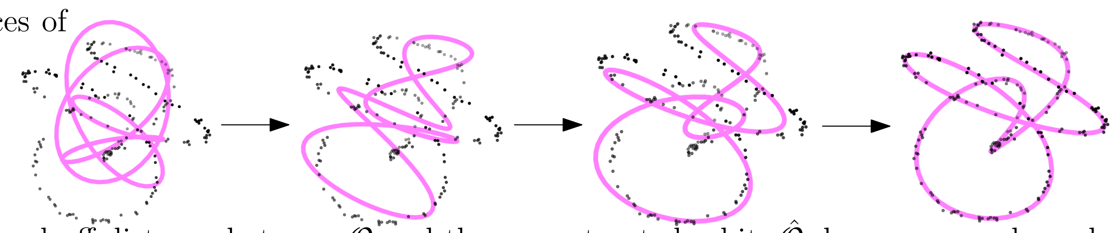
`Step 2`: [J Cahill, DG Mixon, H Parshall, 2023] estimate the normal spaces of $\mathcal{O}$, $N_{x_i}\mathcal{O}$, to calculate the operator

$$\Lambda : M_{n \times n} \to M_{n \times n} \quad \Lambda(A) = \sum_{i=1}^{N} \mathrm{proj}_{N_{x_i}\mathcal{O}} \cdot A \cdot \mathrm{proj}_{\mathrm{span}(x_i)}$$

`Step 3`: optimize, on $Q \in O(n)$, the programs

$$\arg\min \sum_{j=1}^{\dim G} \|\Lambda(QA_jQ^T)\| \quad \text{s.t.} (A_1, \ldots, A_{\dim G}) \in \mathcal{V}_{\mathrm{Lie}}(G, n)$$

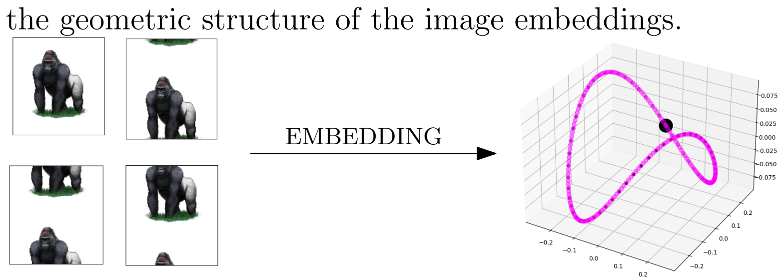
where $\mathcal{V}_{\mathrm{Lie}}(G, n)$ is a list of all derived representations of G in $\mathbb{R}^n$.



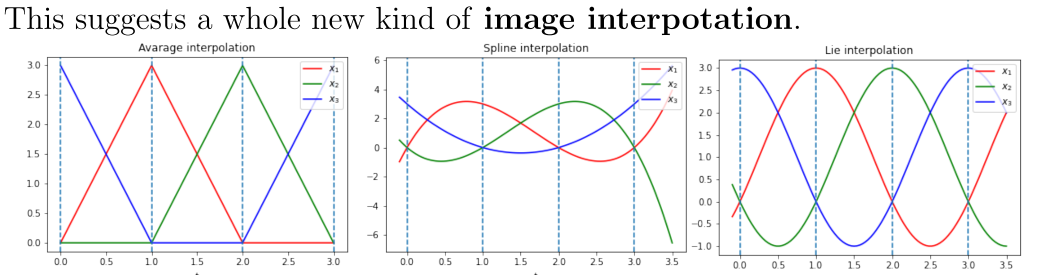
Theorem: if $\{x_i\}_{i=1}^N$ is a sample of the uniform measure on $\mathcal{O}$, then the Hausdorff distance between $\mathcal{O}$ and the reconstructed orbit, $\hat{\mathcal{O}}$, has an upper bound.
Python implementation `https://github.com/HLovisiEnnes/LieDetect`

Applications

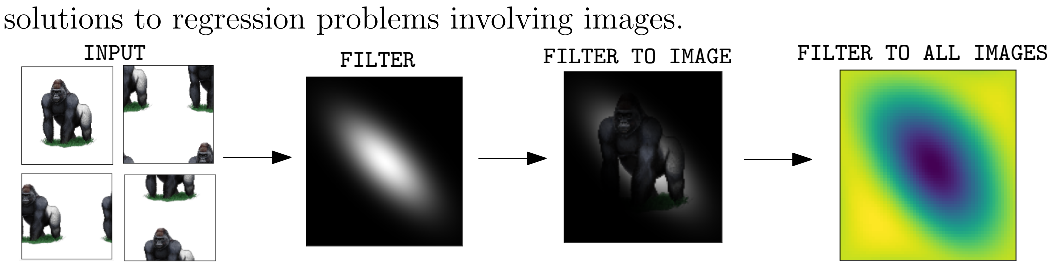
The application of the algorithm allows for the **reconstruction of the orbit**, retriving the geometric structure of the image embeddings.



This suggests a whole new kind of **image interpotation**.



The knowledge of the exact Lie group allows for the application of **abstract harmonic analysis**, a generalization of Fourier analysis that suggests linear solutions to regression problems involving images.



PREDICTIONS