A Tour of Approaches for Automatic Ornament Decomposition

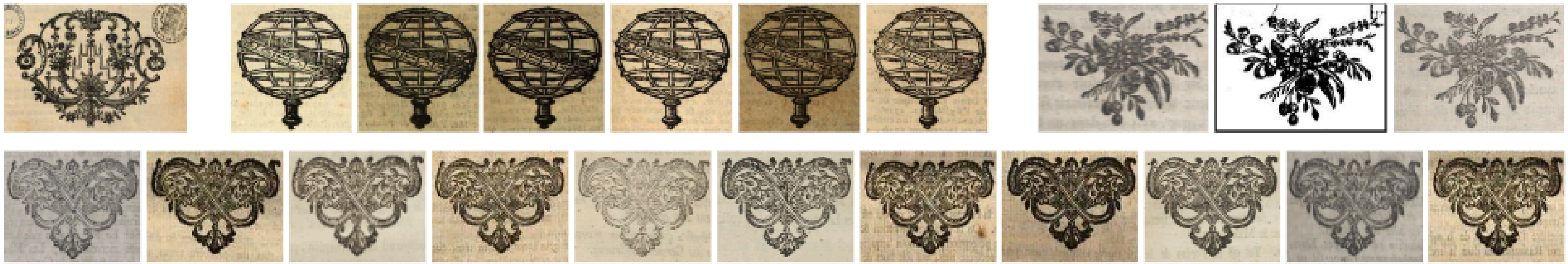
Automatically finding recurrent vignettes in a composed-ornaments dataset

Rémi Emonet / Sayan Chaki
Computer Vision for the Investigation of Ornaments and Ancient Documents

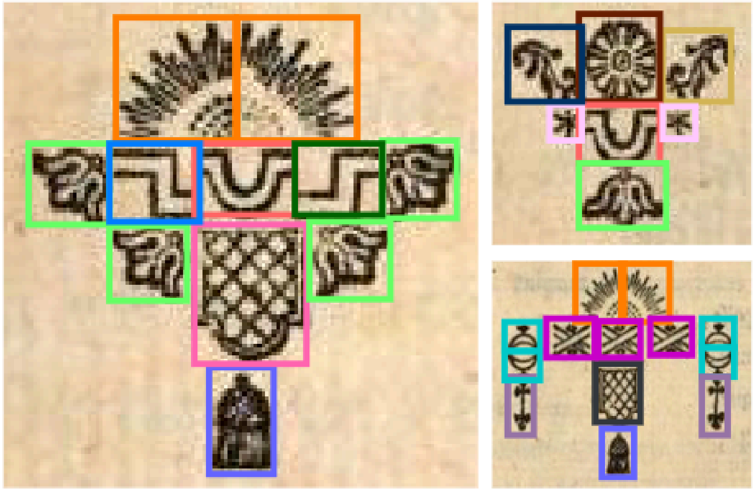
agence nationale de la recherche
Université Jean Monnet Saint-Étienne
ihrim
ROI Rey's Ornament Image investigation
Laboratoire Hubert Curien UMR • CNRS • 5516 • Saint-Étienne
cnrs
institut universitaire de France

Overview

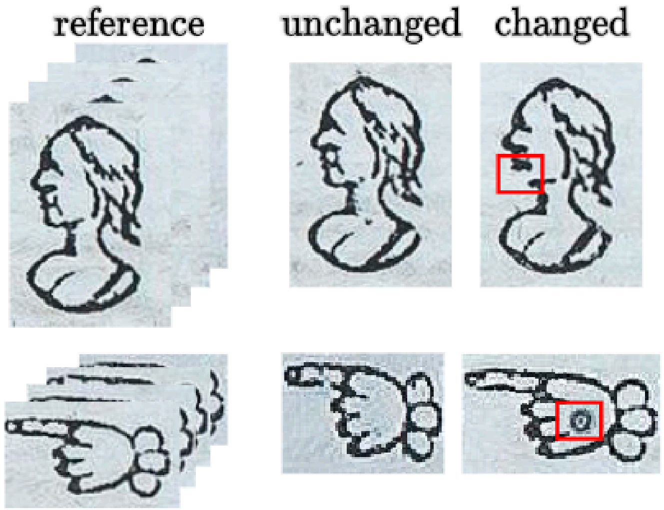
- Part 1
- From Software Engineering to Machine Learning
- Probabilistic Generative Models and Autoencoders
- From Generative Models to Autoencoders
- Part 2
- Building a Synthetic Composite Ornament Dataset
- Methods and Improvements for Unsupervised Object Detection
- Conclusions and Directions



(a) Clustering



(b) Element discovery

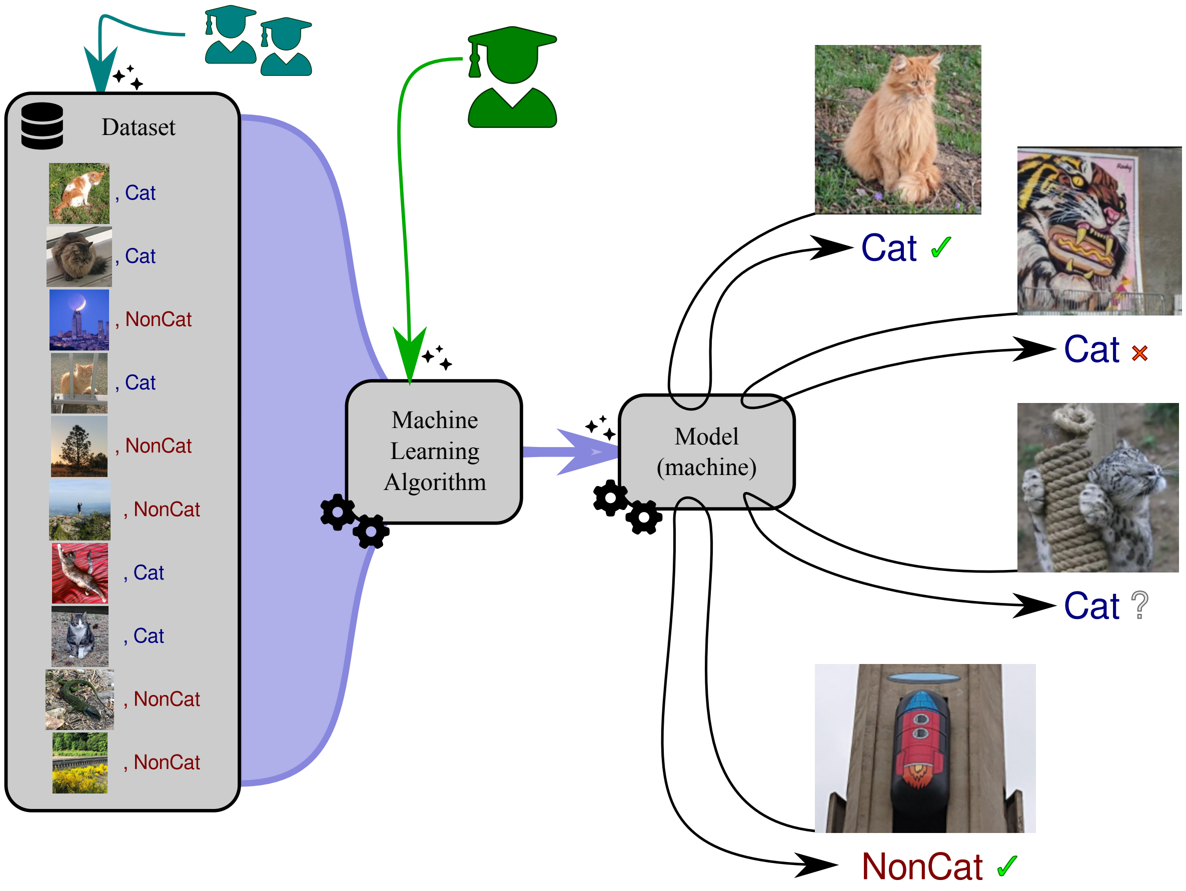


(c) Unsupervised change localization

Part 1

From Software Engineering to Machine Learning

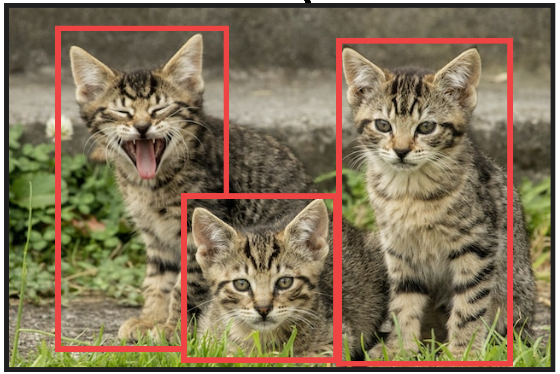
Supervised Machine Learning, illustrated



Example: Object Detection

Learned Model

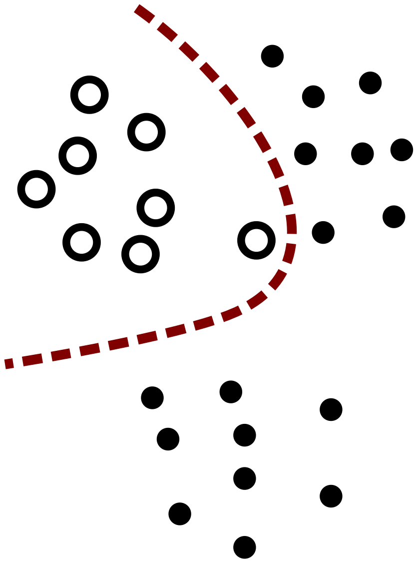
Cat: x=10, y=10, s=2, d=3
Cat: x=60, y=11, s=2, d=2
Cat: x=30, y=25, s=1, d=1



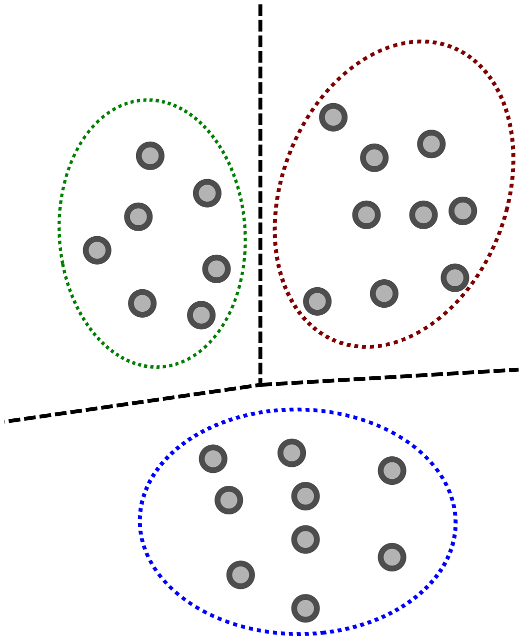
- Predicts object bounding boxes
- Each box has a an objectness (a confidence of existence)
- Each box has a predicted class
- NB: Need labelled training data

Supervised vs Unsupervised |||| ex: Classification vs Clustering

Supervised
(known labels)



Unsupervised
(only inputs)

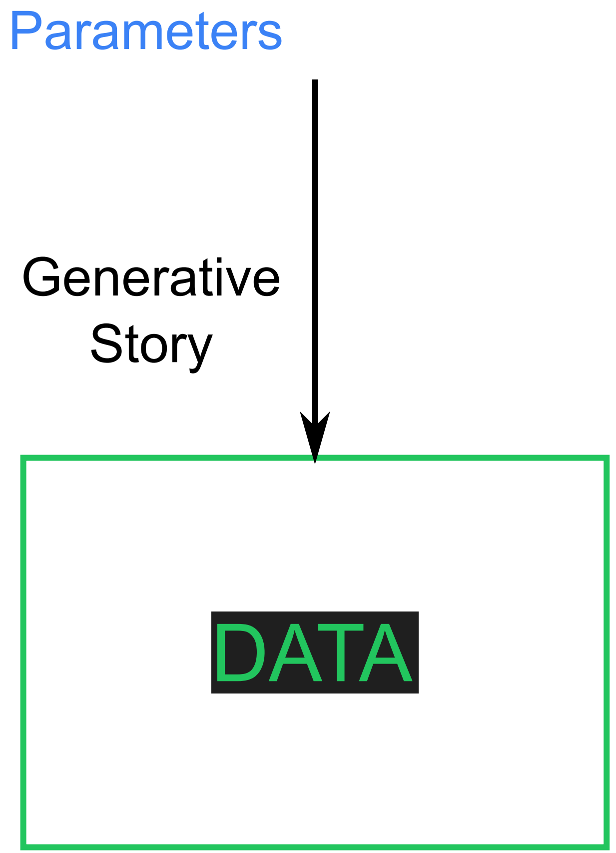


- PCA, UMAP, t-SNE, … any visualization
- Clustering (k-means, spectral, hdbscan)

Probabilistic Generative Models and Autoencoders

Principle of Probabilistic Modeling (generative models)

- Using a generative story / data generation process
  - start with a story about the *data*
  - identify *parameters* and choices in the story
  - use data to select the best parameter values
  - use parameters to do inference, predictions, generation
- Challenges
  - encode constraints/guesses/knowledge as structure
  - derive/write the optimization algorithm



Generative Story: example

Parameters cat(1, 2, 3)

...

...

Generative Story

Renderer

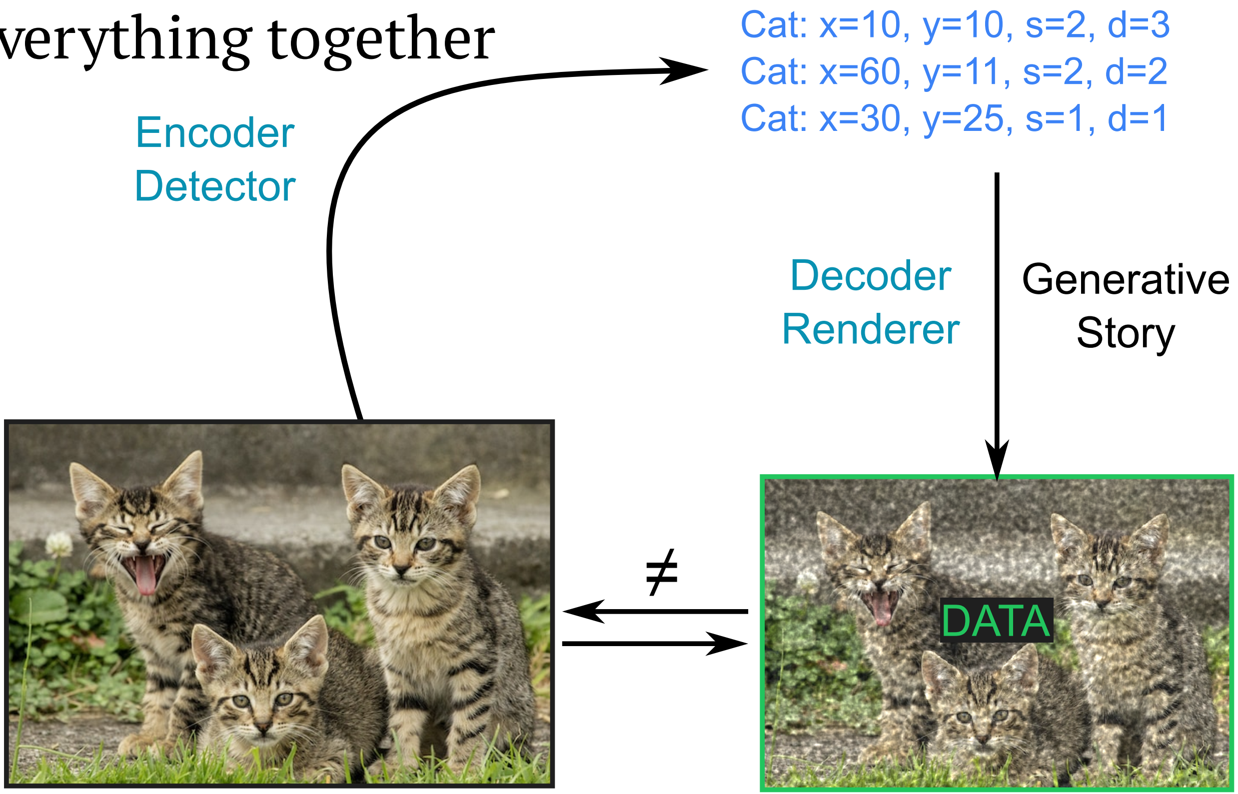
Example of a shelf with cat figurines.

Story:

- decide on a *number of figurines* to use
- for each figurine
  - choose *one size* among the 5 sizes
  - place the figurine at a *random position*
  - take a *picture*

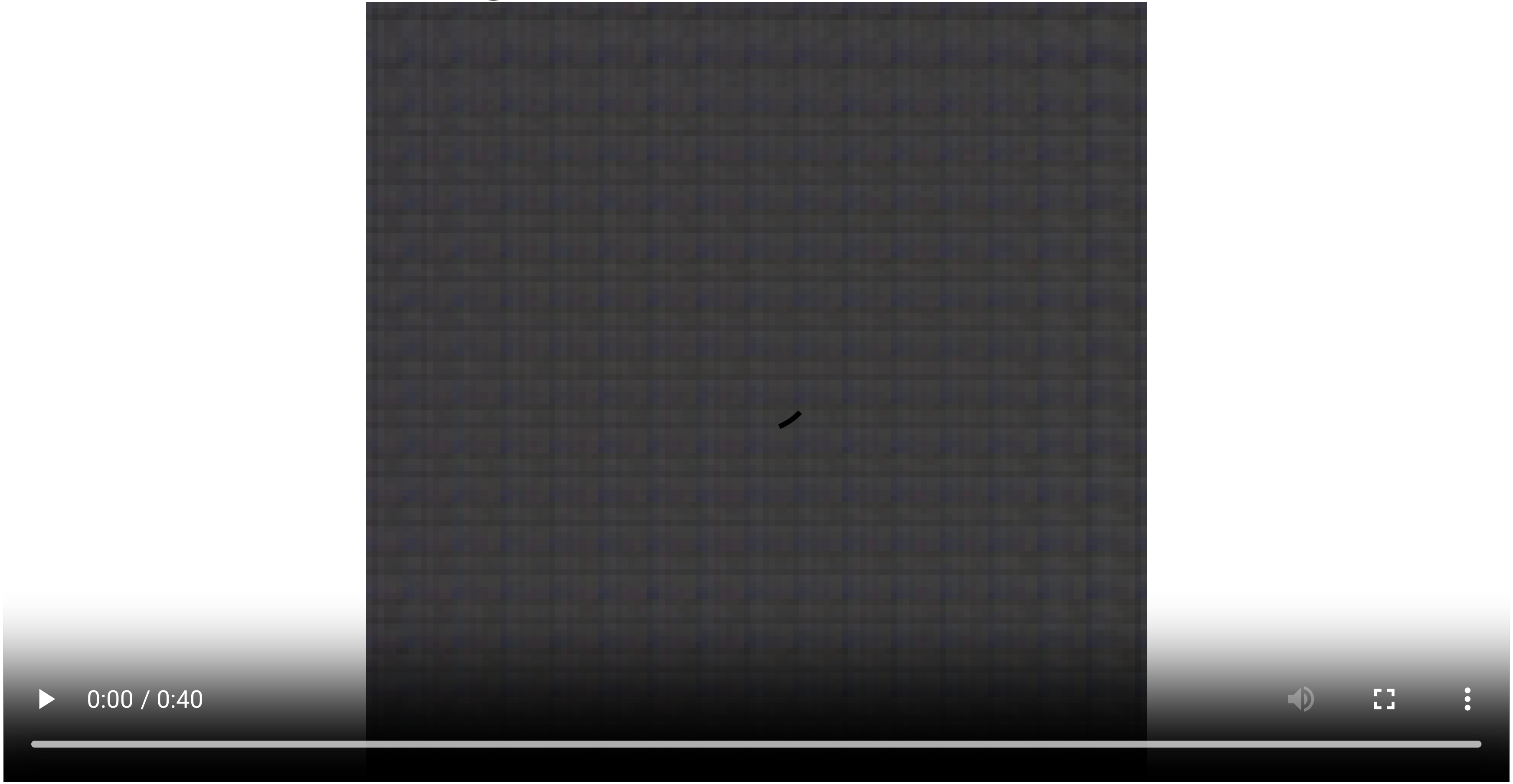
From Generative Models to Autoencoders

Bringing everything together



- Encoder/Decoder = Detector/Renderer
- Train to encode and properly reconstruct
- Self-supervised (non-supervised) training

Teaser: Autoencoding Ornaments



Part 2

Building a Synthetic Composite Ornament Dataset

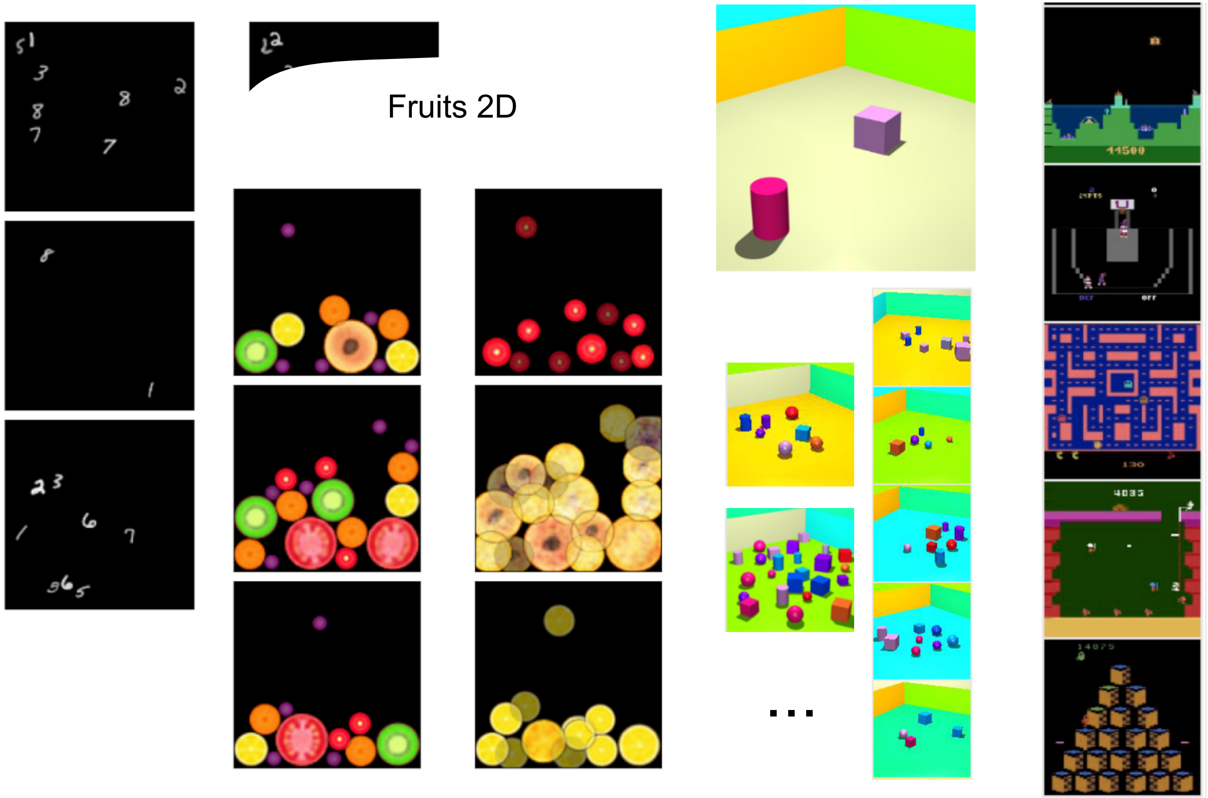
Existing Datasets

MultiMNIST

Fruits 2D

3D Rooms

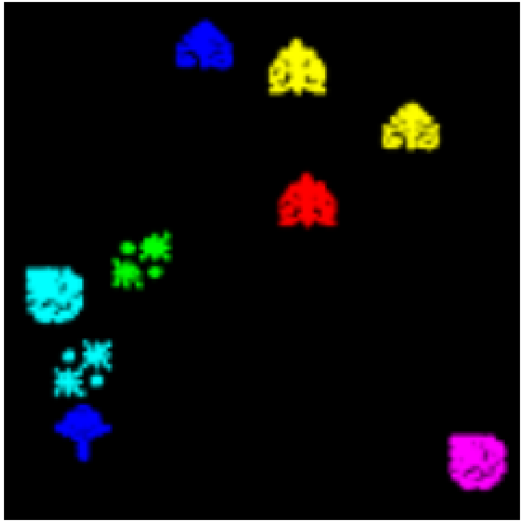
Atari Games"



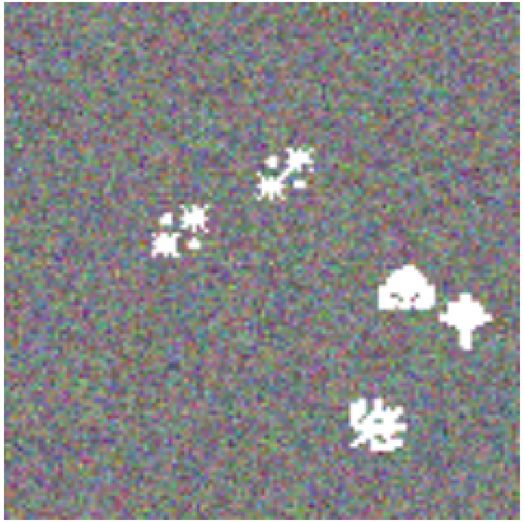
Custom Ornaments Datasets

Scattered Vignettes

Colorful Vignettes



With Background



- Variations
  - size
  - rotation
  - color ⇔ shape association
- Differences between "training" and "test"?
  - same vignettes?
  - same colors
  - same scale

Methods and Improvements for Unsupervised Object Detection

Rough Classification of Existing Unsupervised Approaches

Attend Infer Repeat

- AIR, SPAIR
- DAIR, SQAIR, R-SQAIR
- …

Mixture of layers

- IODINE, GENESIS, …

Joint models

- SPACE
- RICH
- …

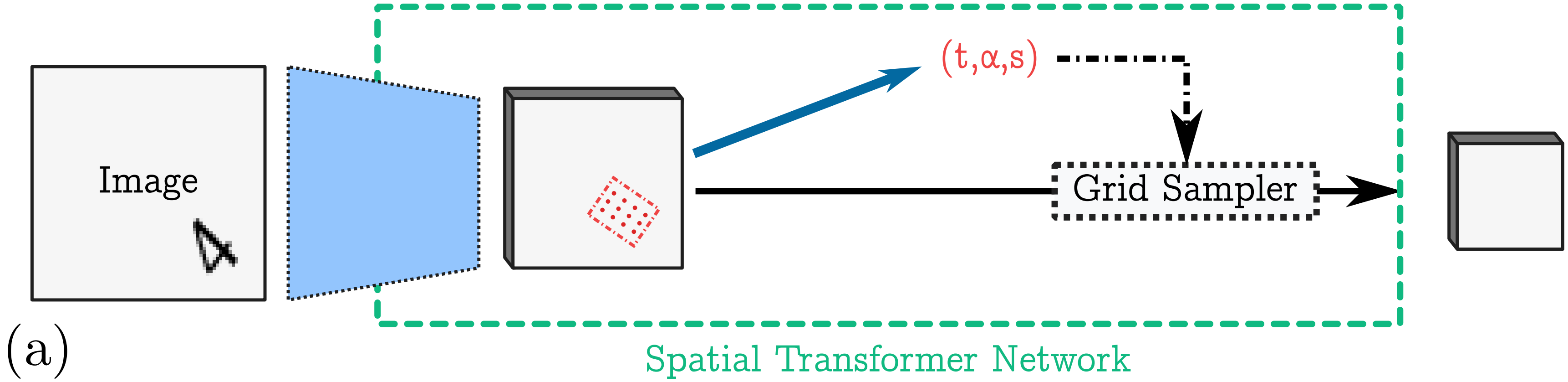
Robustness to transformations

- Learned
  - Spatial Transformer Networks (STN)
  - canonicalization
- Built-in
  - Group-equivariant networks
  - Fully-equivariant networks?
- Mixed

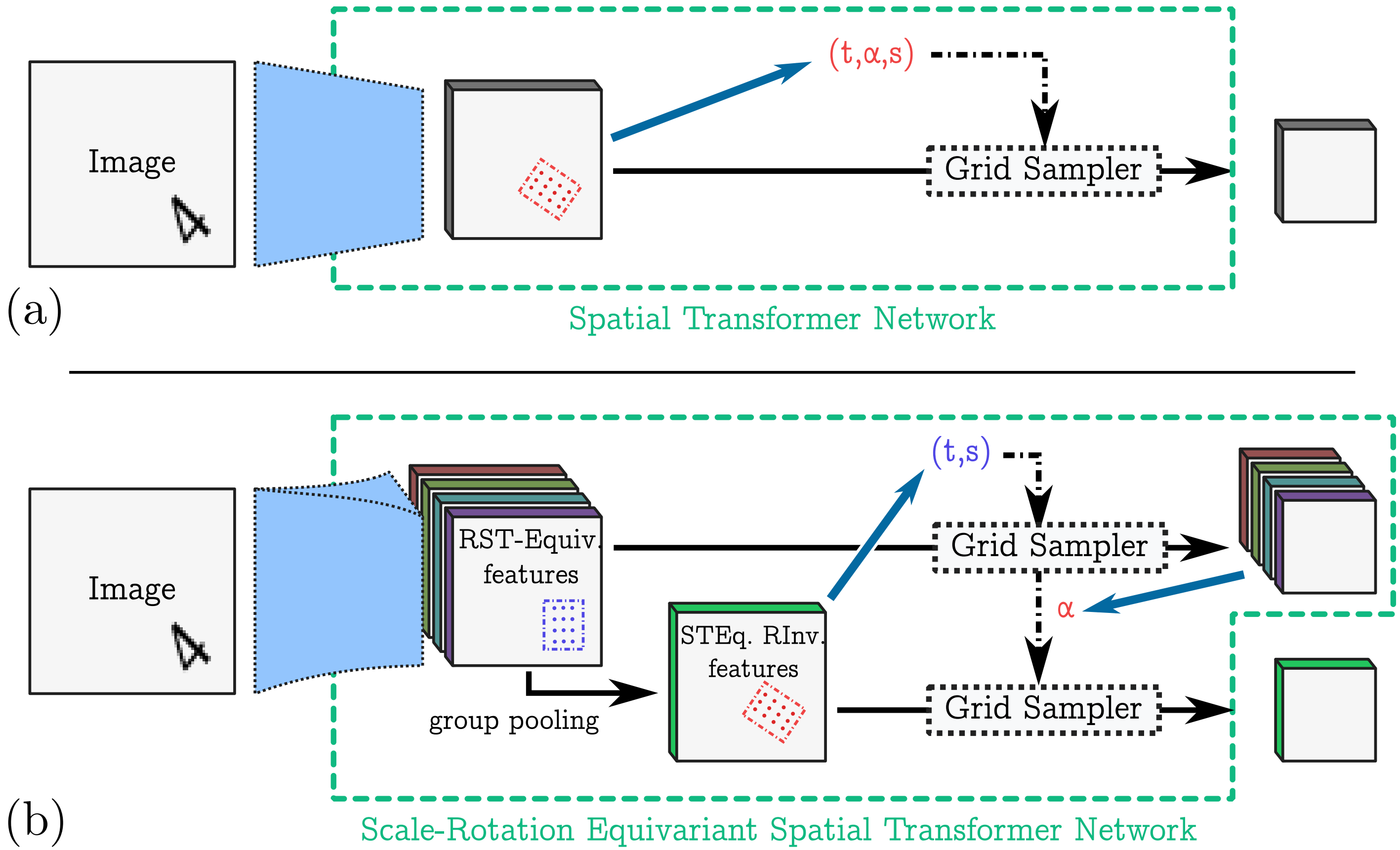
Spatial Transformer Networks (STN)

Spatial Transformer Networks

- predict a transformation
- extract a glimpse (transformed patch)
- do further processing on this glimpse
- fully differentiable process

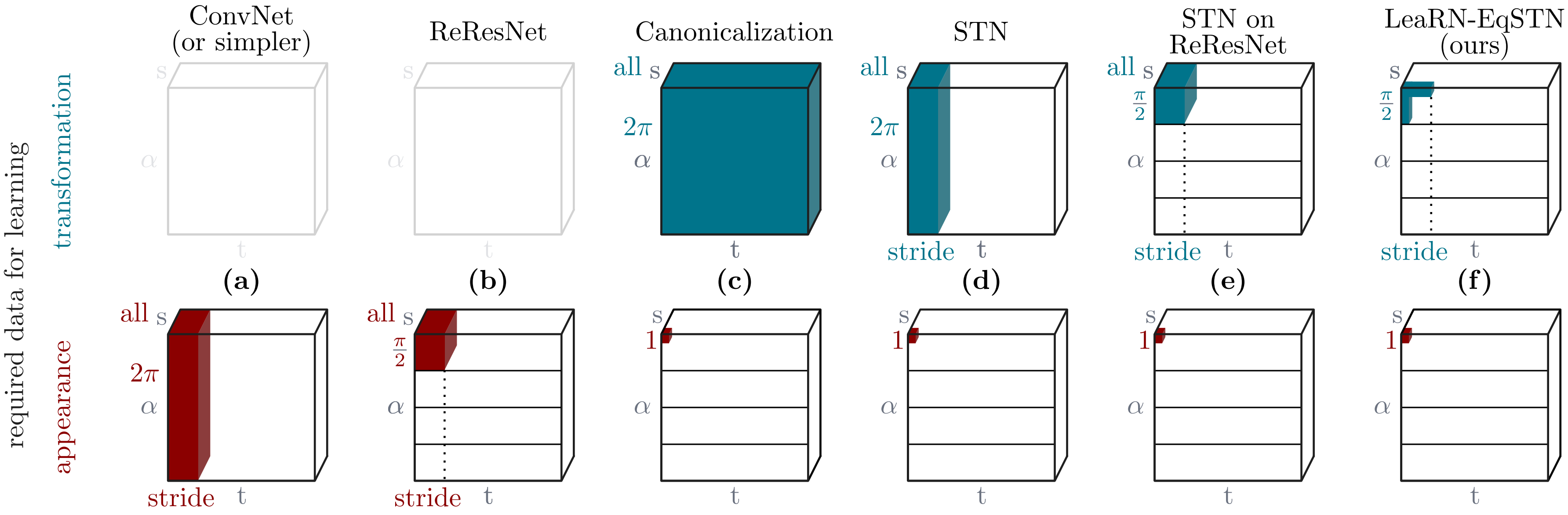


Improving STN Data-Efficiency by Sequential Estimation

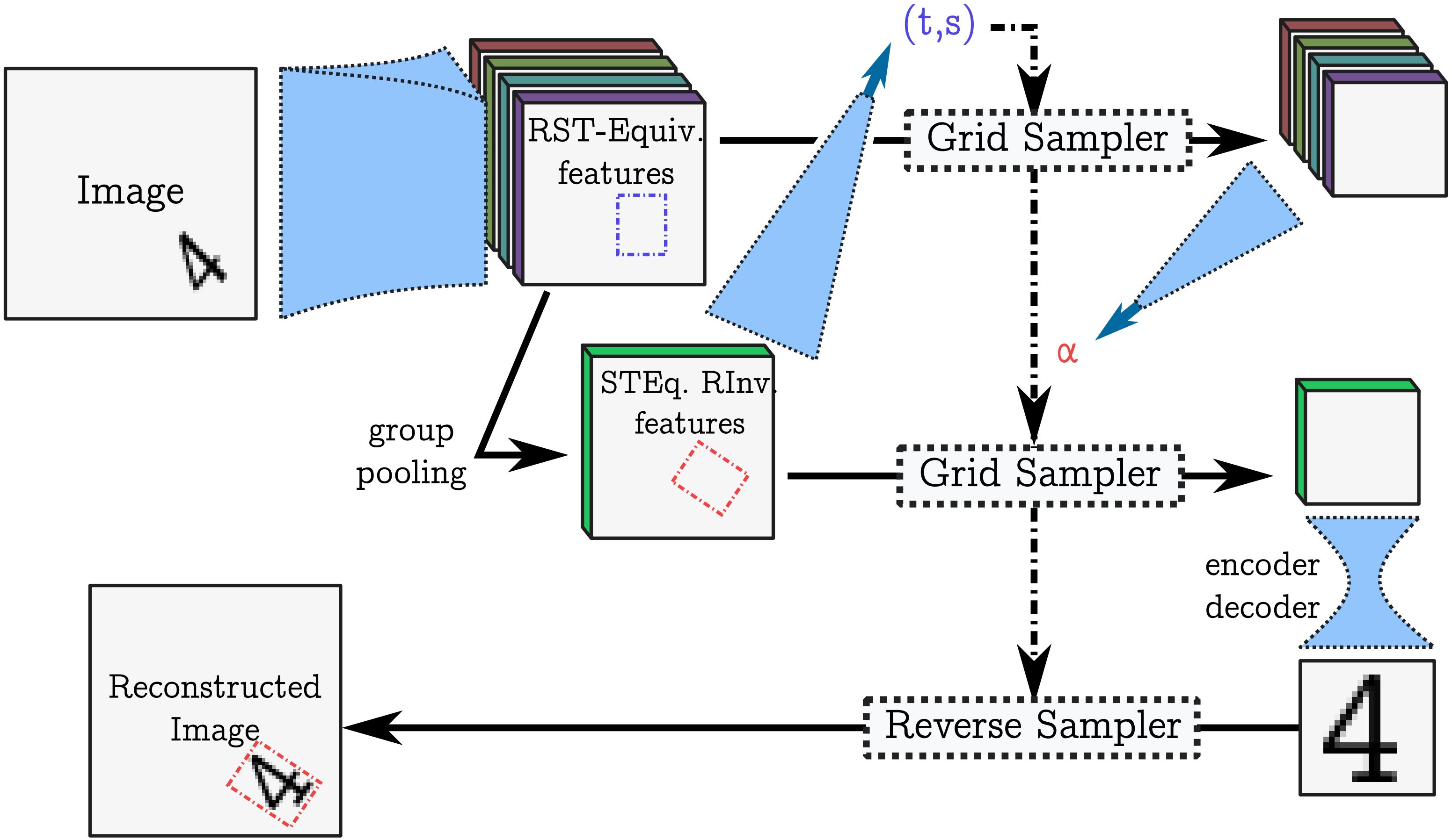


Details on Data Requirements

Goal: translation (t), rotation (α), scale (s) equivariance



Auto-Encoding Using STN



Technical Elements

- Sequential estimation of composite STN
- Custom Learnable Riesz network
- CNN-Based Glimpse decoder

Some Experiments

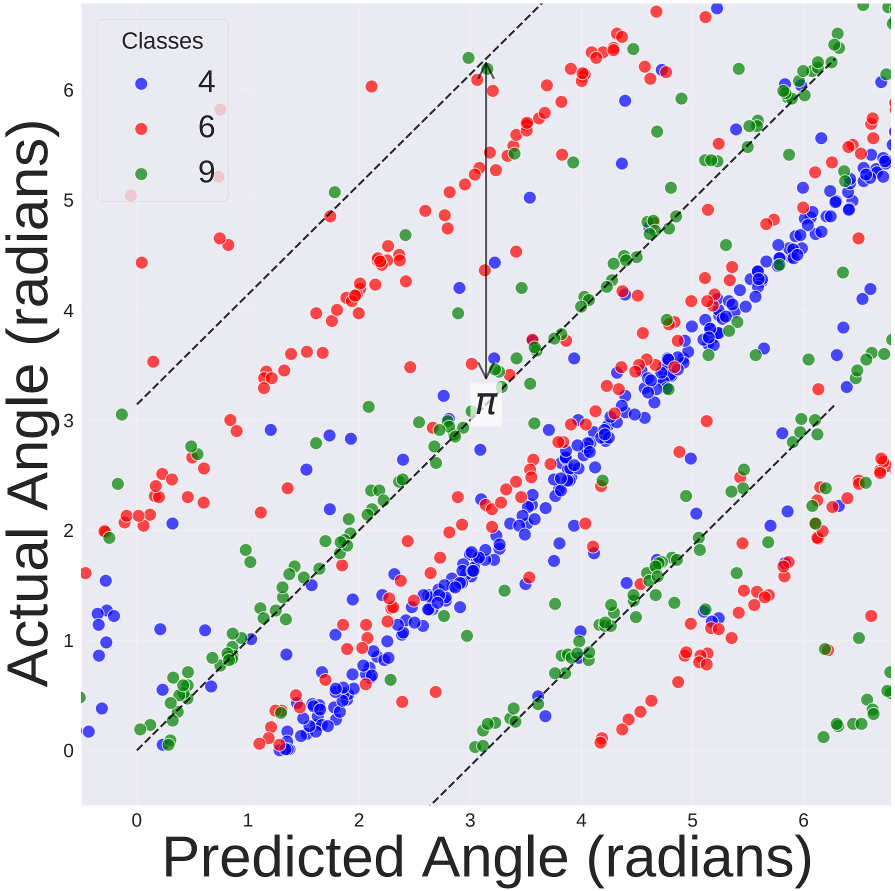
Typical test case

- synthetic dataset so we can evaluate
- mixture of rotated, scaled, MNIST (digits), here digits 4, 6 and 9

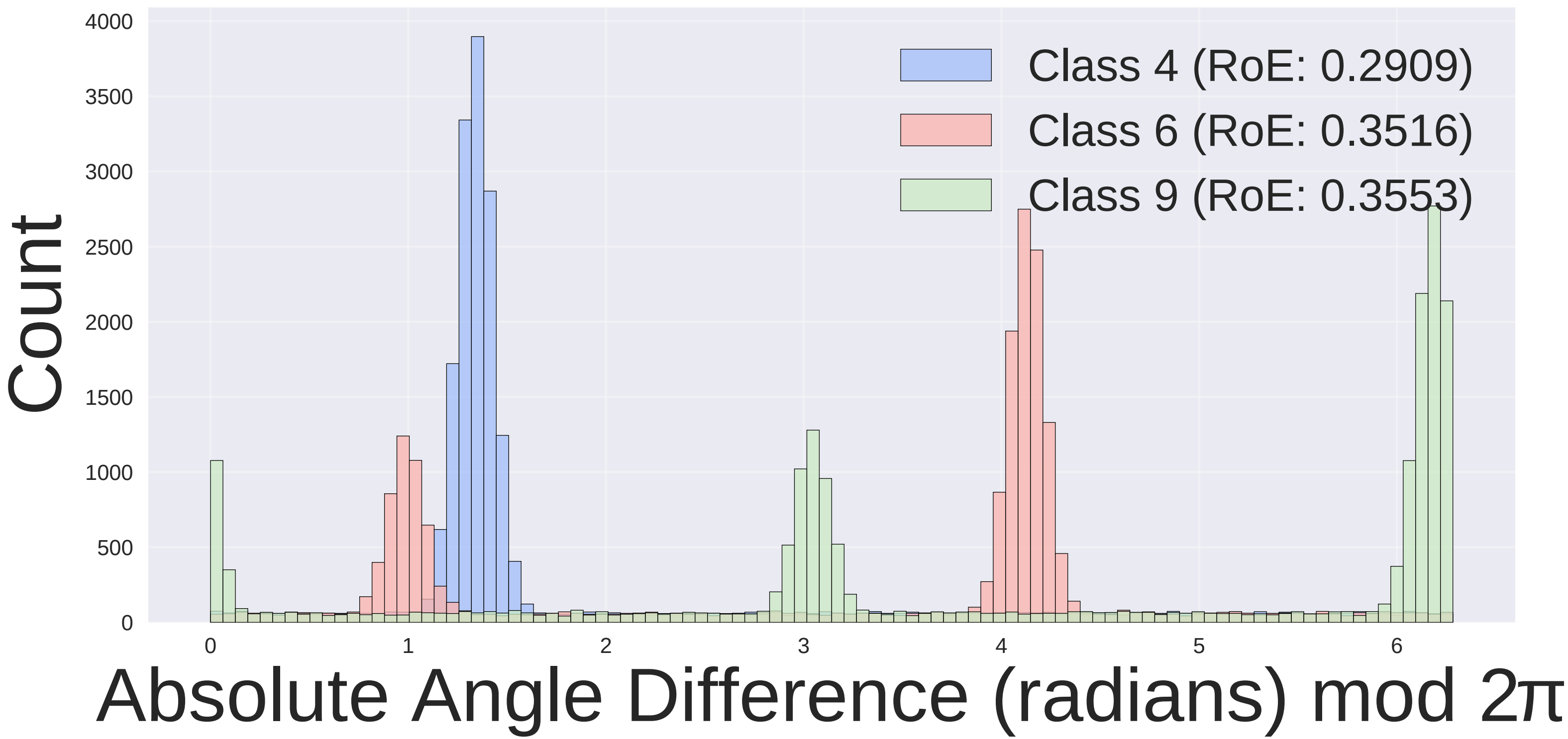
Evaluation goal

- can we cluster the digits
- can we recover the rotation (position, and scale etc)

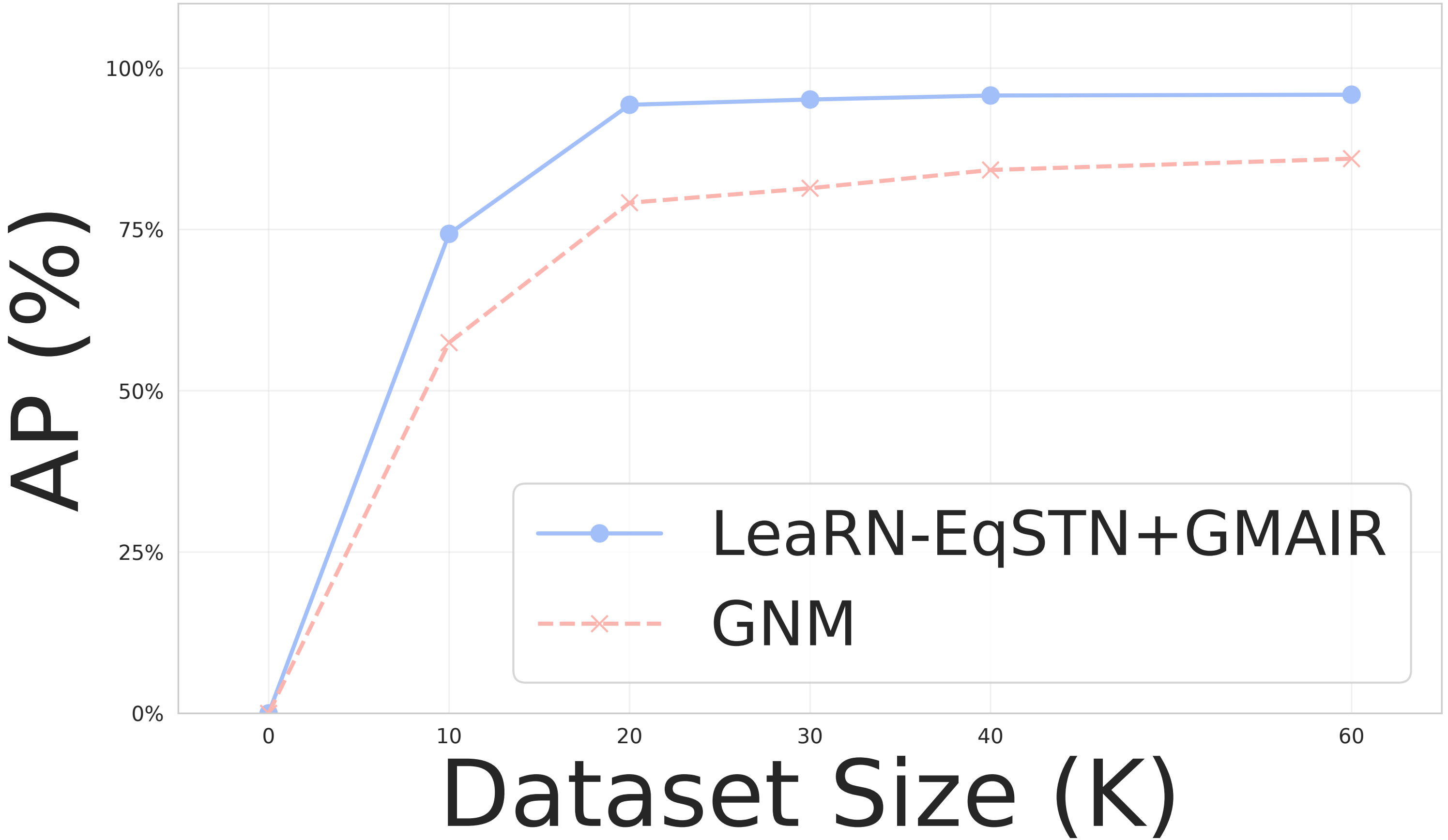
...


Classes
4
6
9
π
Actual Angle (radians)
Predicted Angle (radians)
0
1
2
3
4
5
6

…


Class 4 (RoE: 0.2909)
Class 6 (RoE: 0.3516)
Class 9 (RoE: 0.3553)
Count
0
500
1000
1500
2000
2500
3000
3500
4000
0
1
2
3
4
5
6
Absolute Angle Difference (radians) mod 2π

...



100%
75%
50%
25%
0%
AP (%)
0
10
20
30
40
60
Dataset Size (K)
LeaRN-EqSTN+GMAIR
GNM

Conclusions and Directions

Conclusions and Directions

- Summary
  - a generative story of composed ornaments
  - an autoencoder approach
  - a semantic-rich latent representation
  - targeting data-efficient learning
    - invariance to rotation and scale
    - sequential estimation

- Open questions
  - accuracy ⇔ automation tradeoff
  - better appearance model (flow matching)
  - learning the composition style
    - learning the similarity
    - suggesting "synonyms" of vignettes

- Thank you!