TEXAS
The University of Texas at Austin

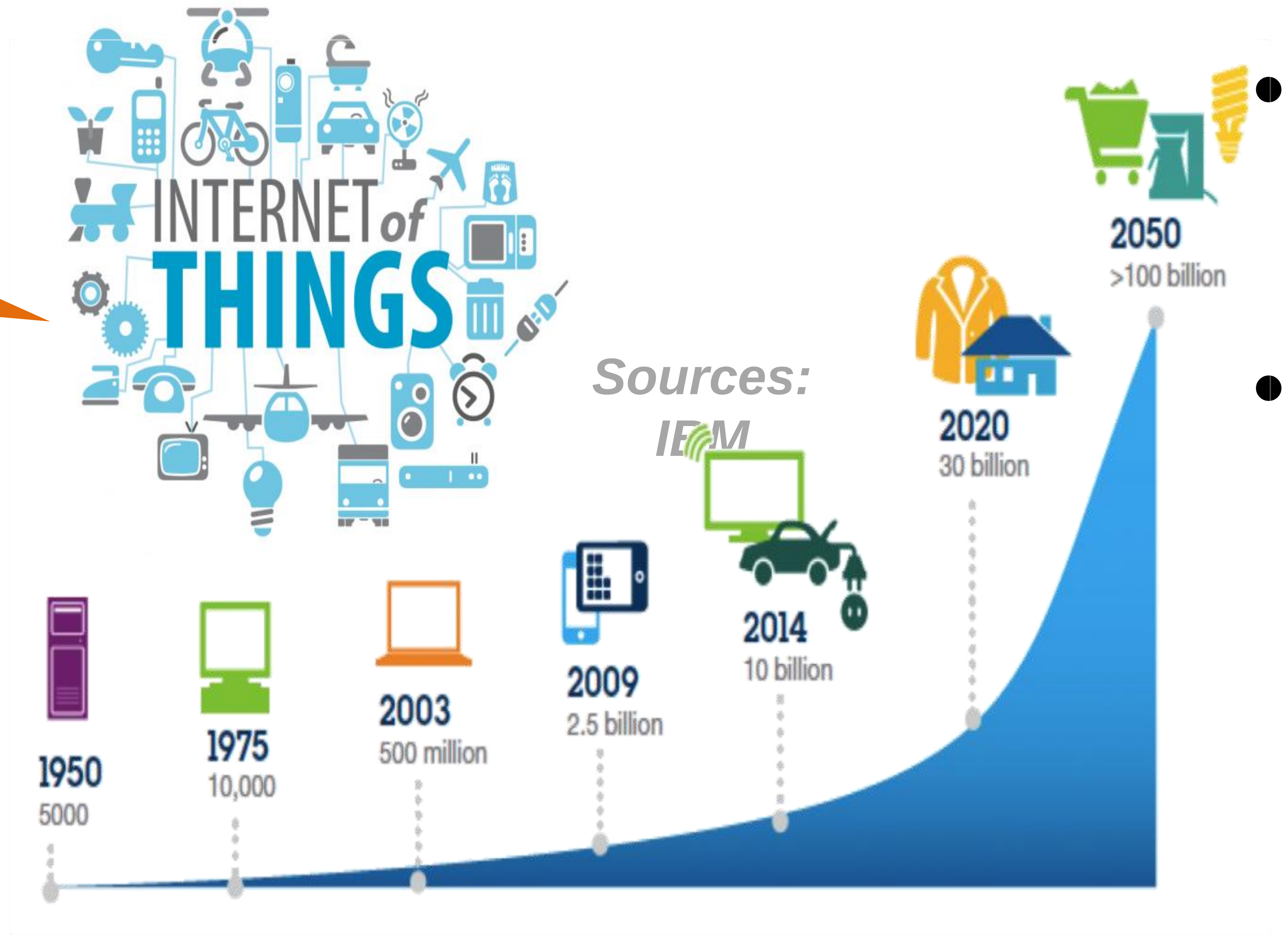
GeniusRoute: A New Analog Routing Paradigm Using Generative Neural Network Guidance

Keren Zhu, Mingjie Liu, Yibo Lin, Biying Xu, Shaolan Li, Xiyuan Tang, Nan Sun and David Z. Pan
ECE Department
The University of Texas at Austin
This work is supported in part by the NSF under Grant No. 1704758, and the DARPA ERI IDEA program

Outlines

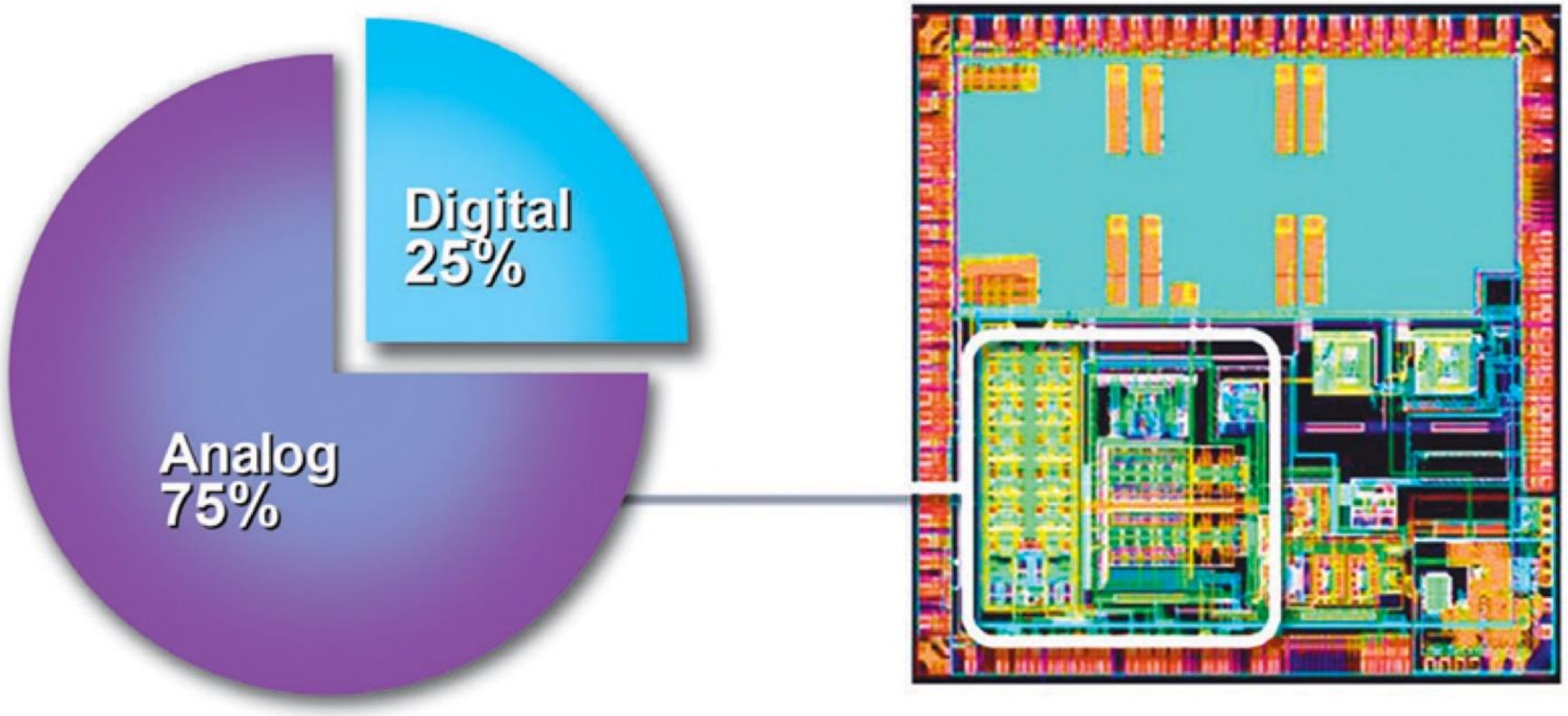
- Introduction and Problem Formulation
- GeniusRoute Framework
- Experimental Results
- Conclusion

High Demand of Analog/Mixed-Signal IC



- Anything related to sensors needs analog!
- Internet of Things (IoT), autonomous and electric vehicles, communication and 5G networks…

A Bottleneck in IC Design: Analog/Mixed-Signal

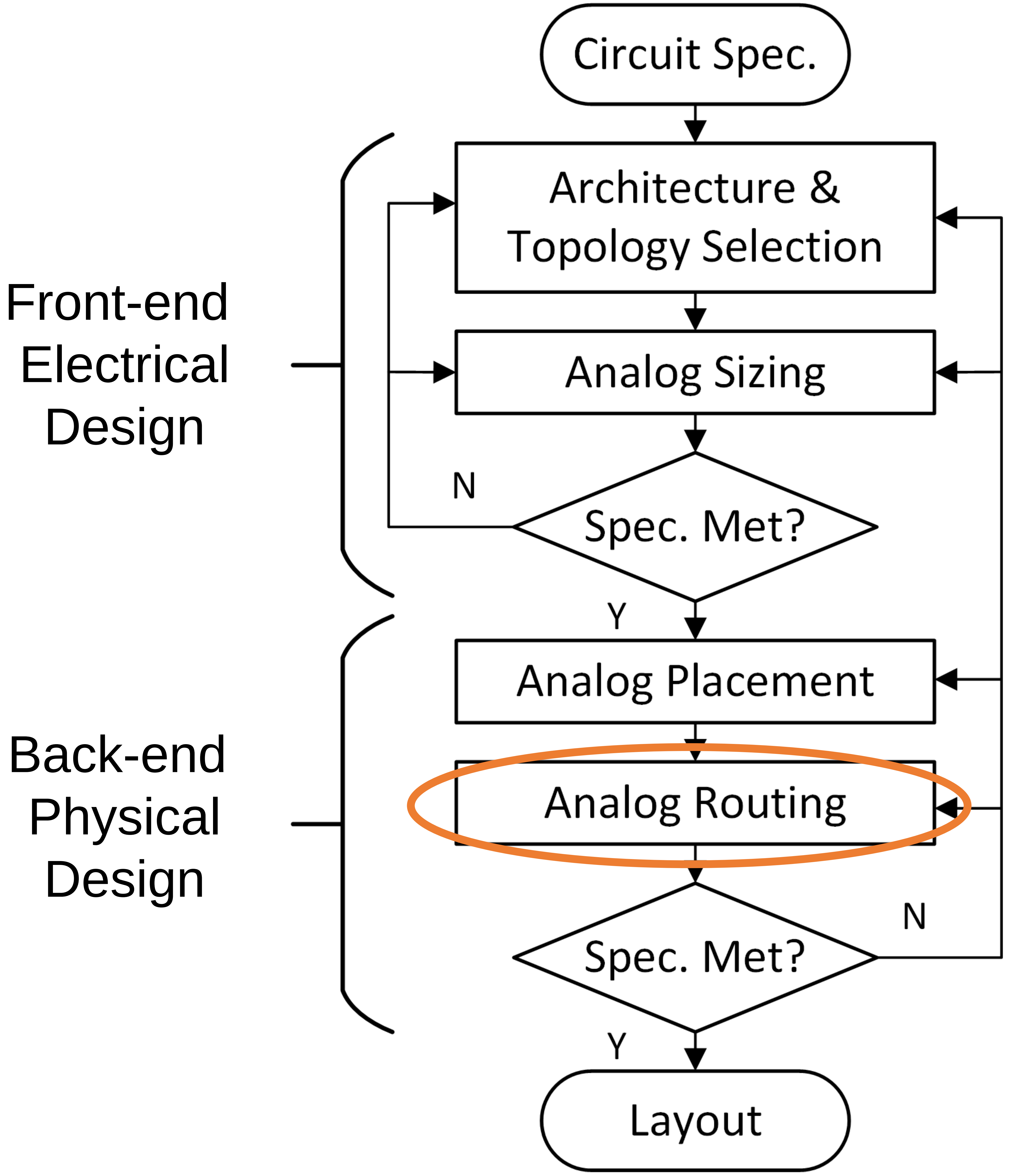


Analog parts of IC take large design efforts

A major reason: analog circuit layout is usually done manually

[IBS and Dr. Handel Jones, 2012]

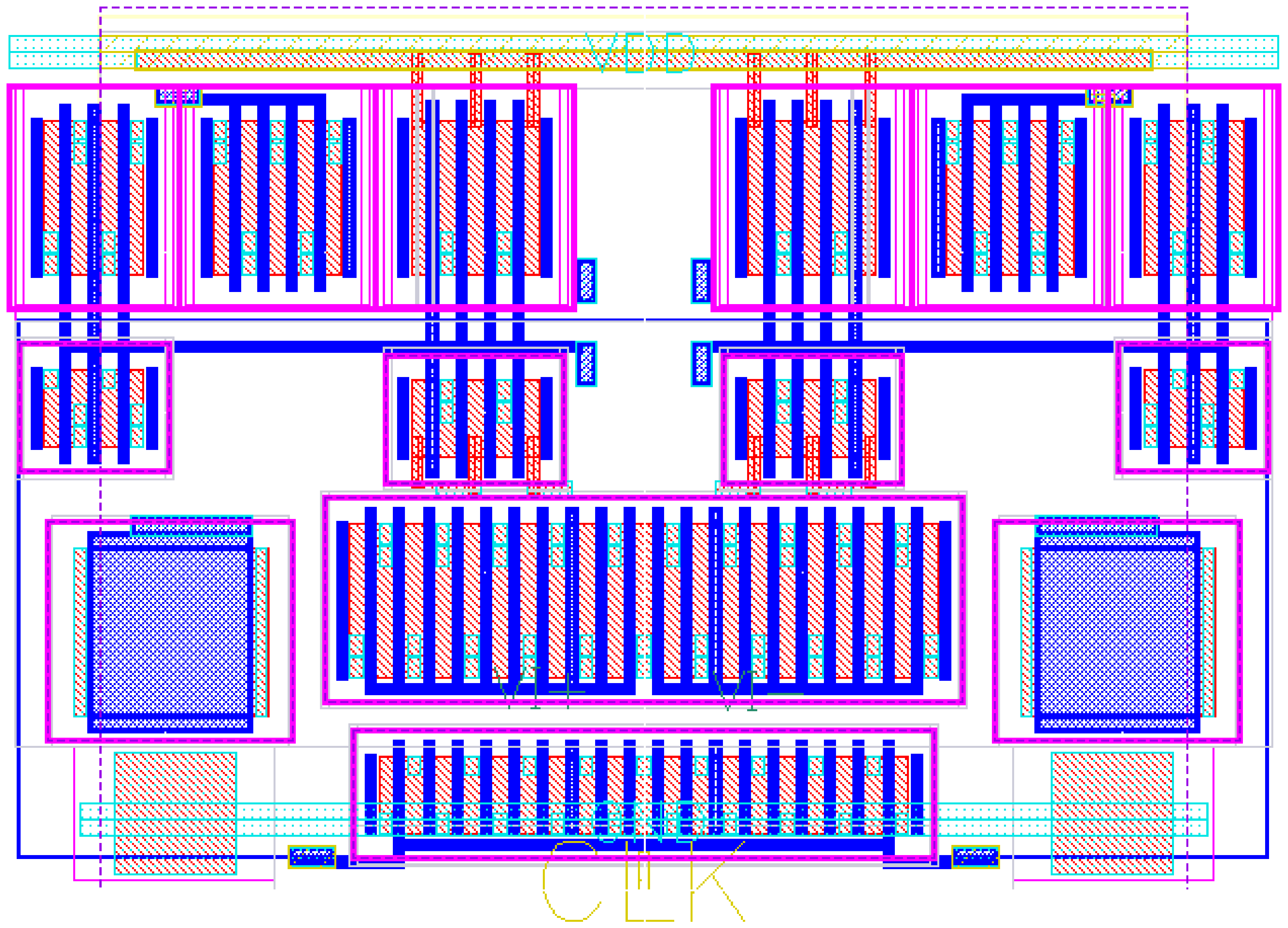
Typical Automatic Analog Circuit Design Flow



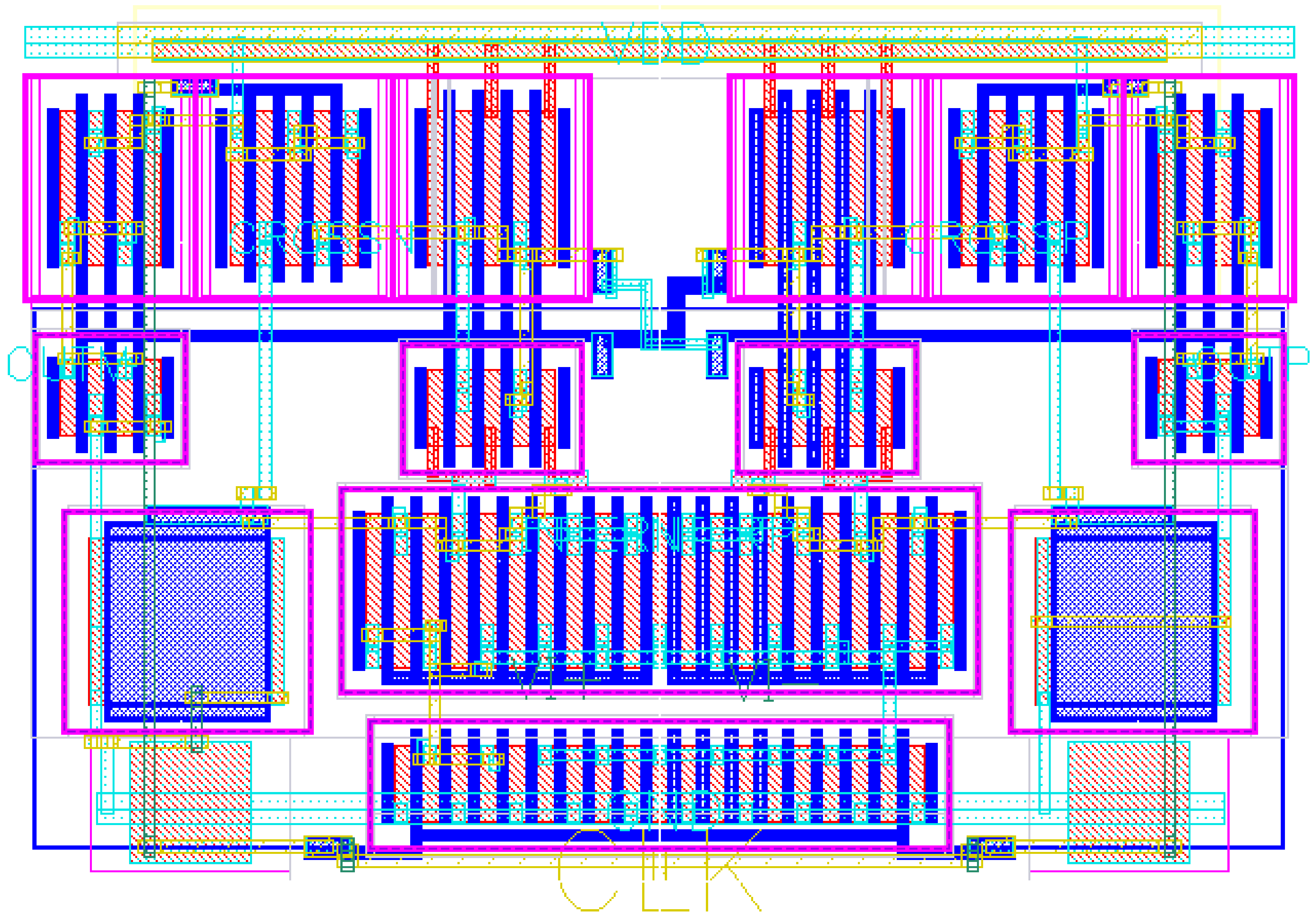
- Automated analog design often consists of front-end and back-end flows

- Physical design (back-end) is separated in placement and routing

Analog Routing Problem

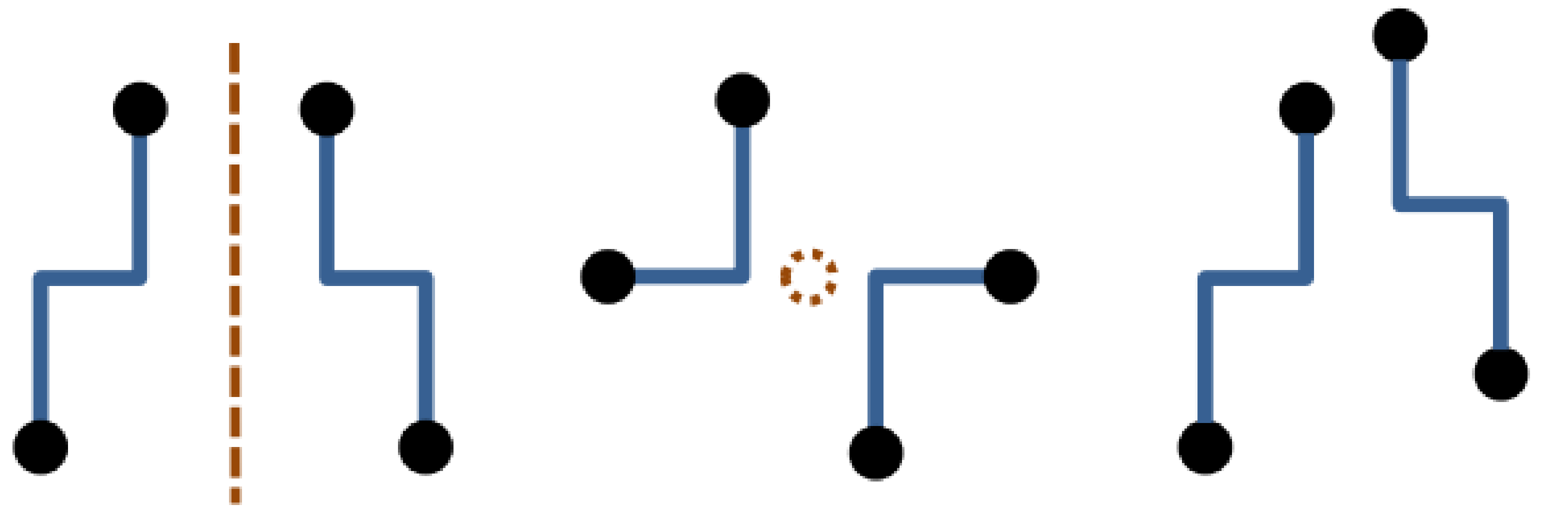


Placement



Routed Layout

Challenges in Formulating Analog Routing Problem



[Ou et al., 2014]

Shielding,
Avoid active region,
…

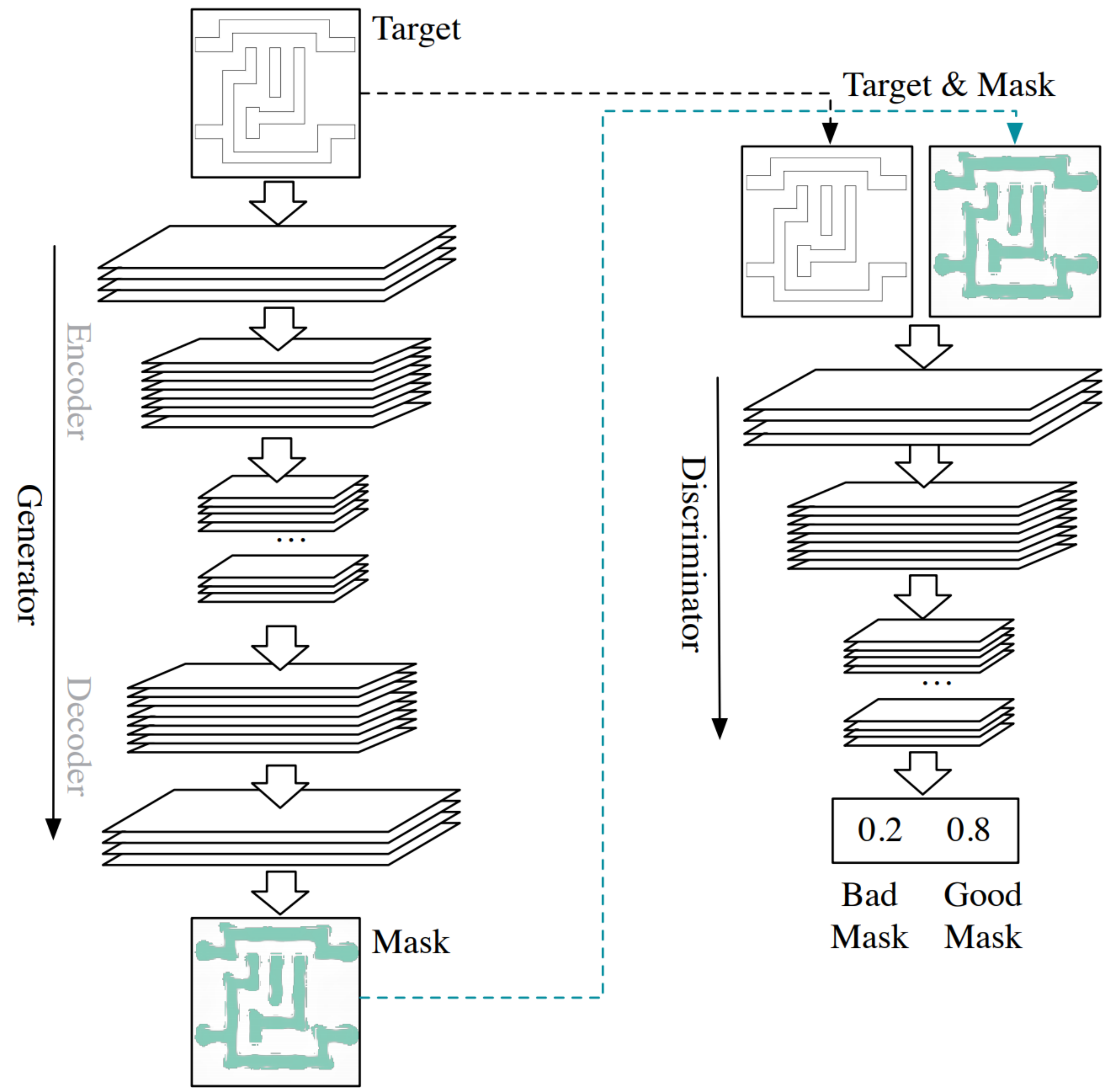
Symmetry constraints are widely accepted

No standard rule for additional constraints. Design-dependent.

Automatically learn from human layouts?

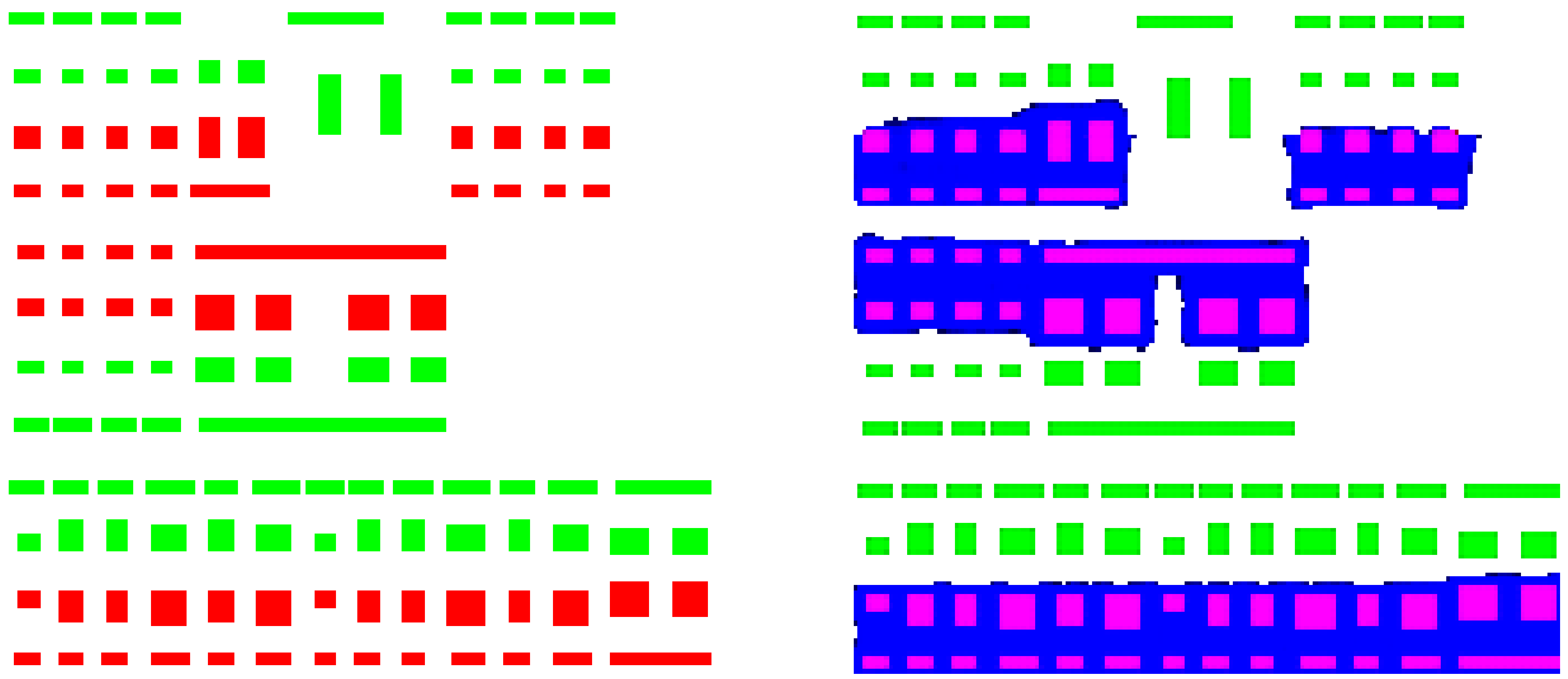
Emerging Machine Learning Applications

Lithography: GAN-OPC



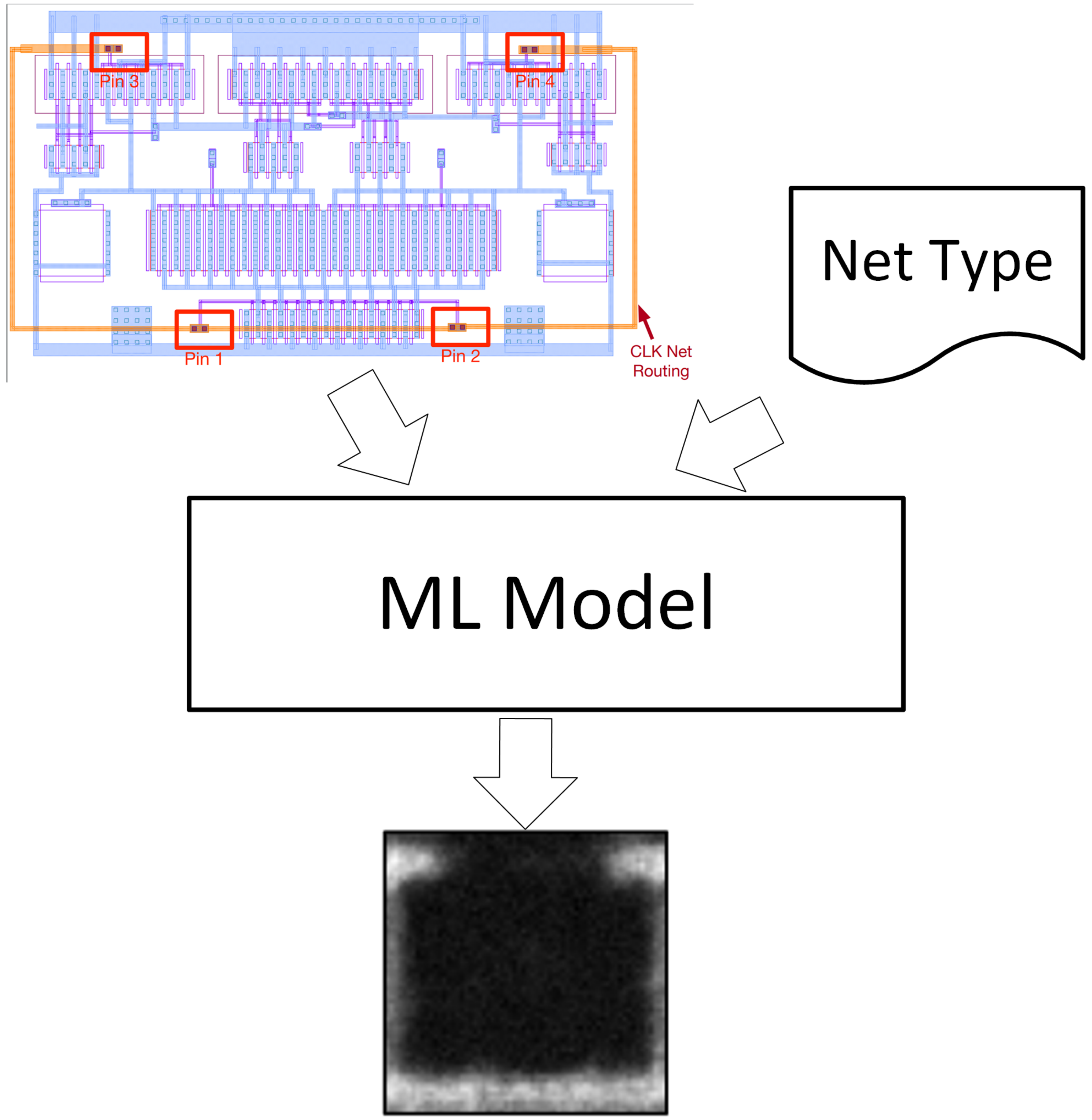
[Yang et al., 2018]

Physical Design: WellGAN



[Xu et al., 2019]

Automatically Learn Guidance from Human Layouts



- Learn routing guidance
  - Where the human would likely to route the nets
- Extract training data from labeled layouts
- Apply learned model to automatic routing as guidance

A ML-Guided Routing Problem

Heuristic constraints: use a set of detailed heuristics as routing constraints

Placement → **Explicated Constraints** → **Routing**

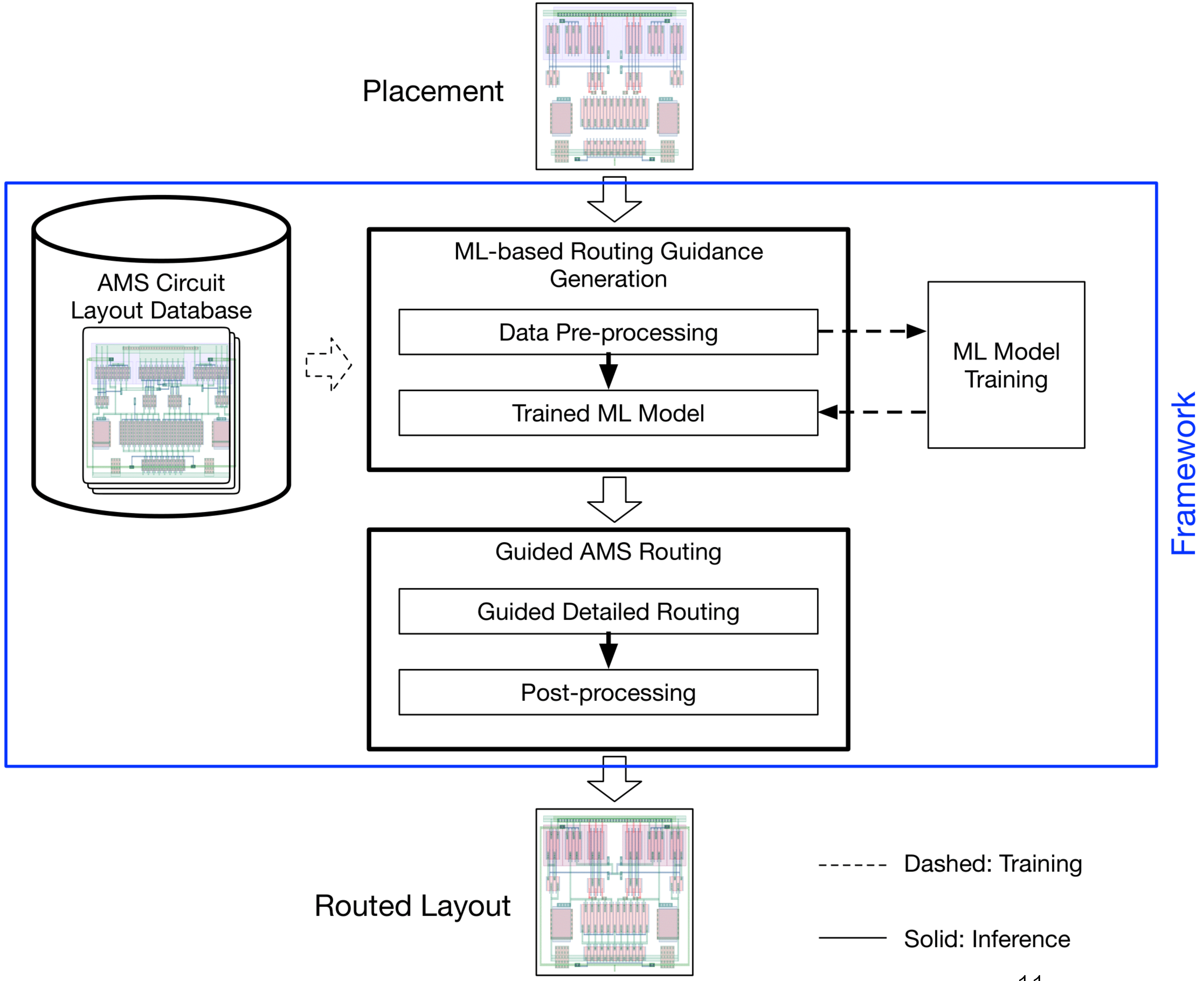
Conventional Approach

Routing guide: routing strategies learned from human

Placement → **Symmetric Constraints + ML-based Routing Guide** → **Routing**

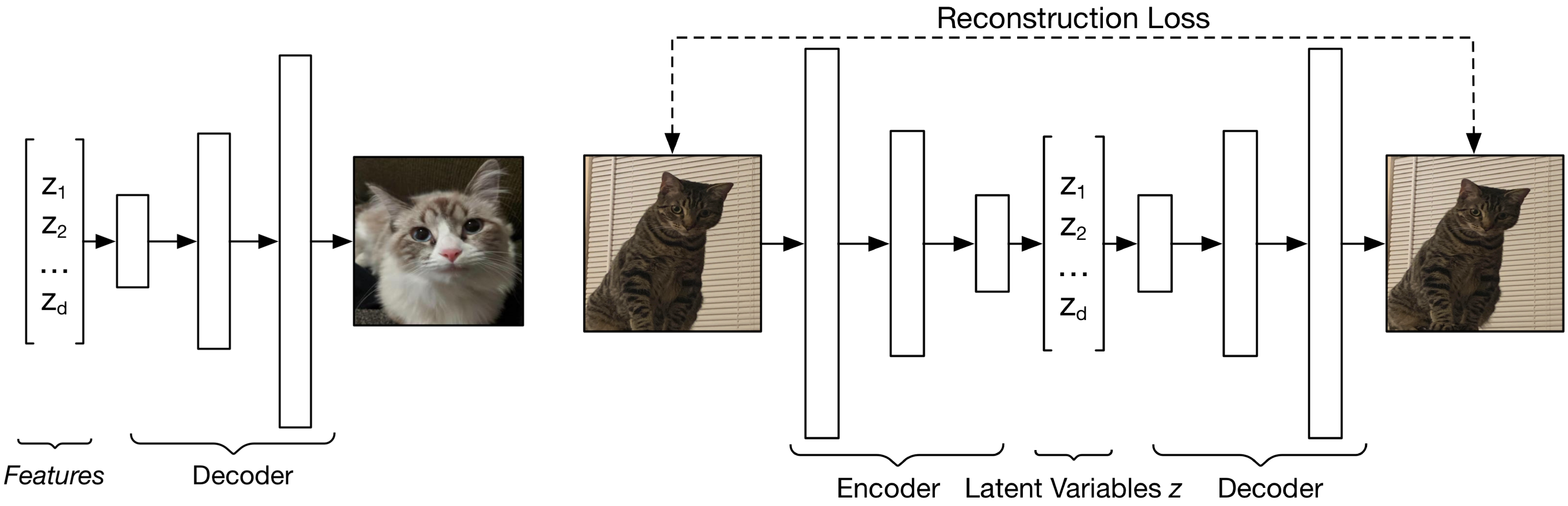
GeniusRoute Approach

The GeniusRoute Flow

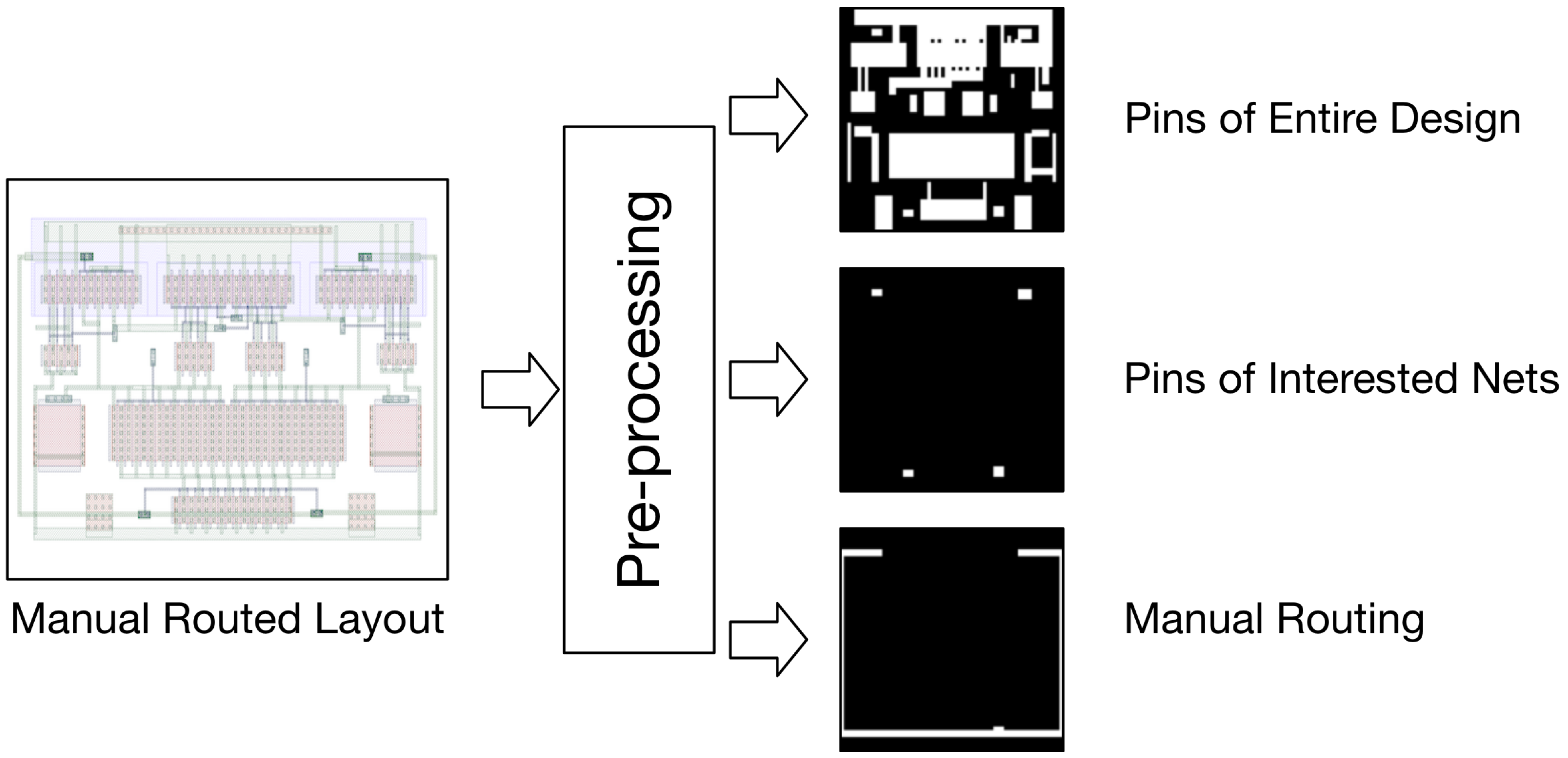


- Learn from GDS layouts
- Pre-process layouts into images
- Predict routing probability using autoencoder
- Use prediction as detailed routing guidance

Generating Images with Generative Neural Network



Data-Preprocessing: Extracting Routing from Layouts

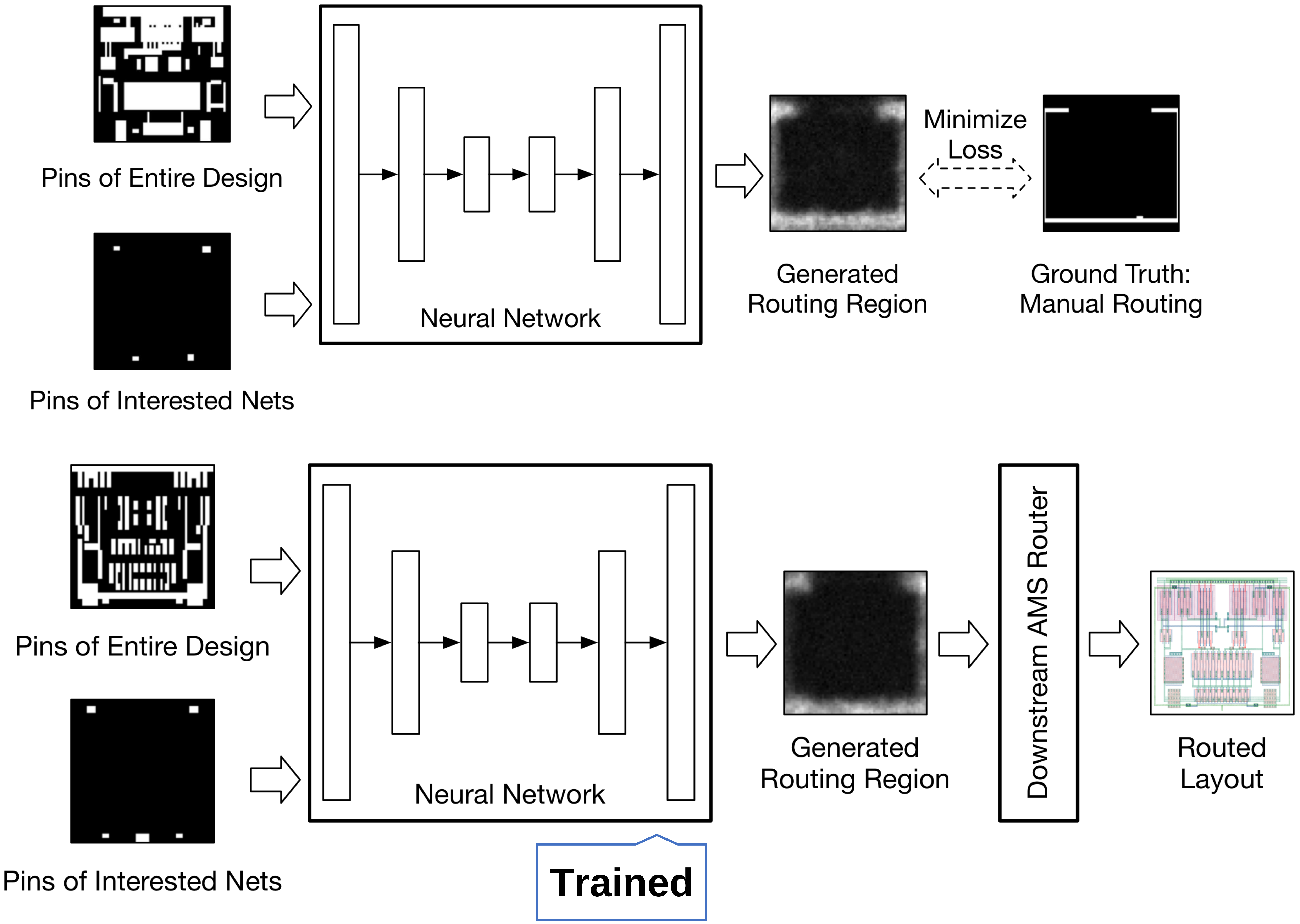


Extract “pins” and routing of nets

Three categories of models:

- Symmetric nets
- Clocks
- Power and Ground

GeniusRoute: Learning Routing Patterns from Human

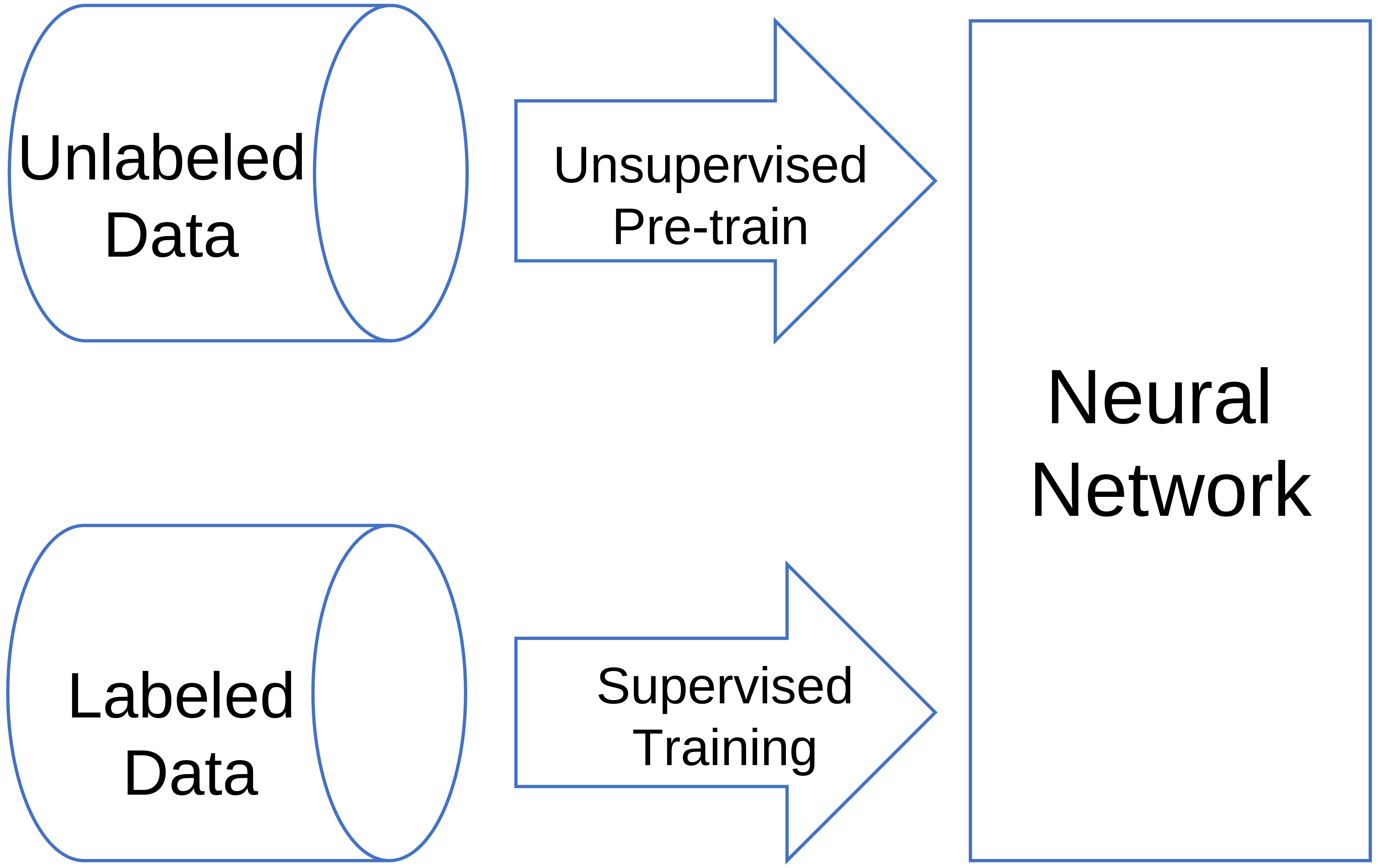


Training Phase

Do we have enough data?

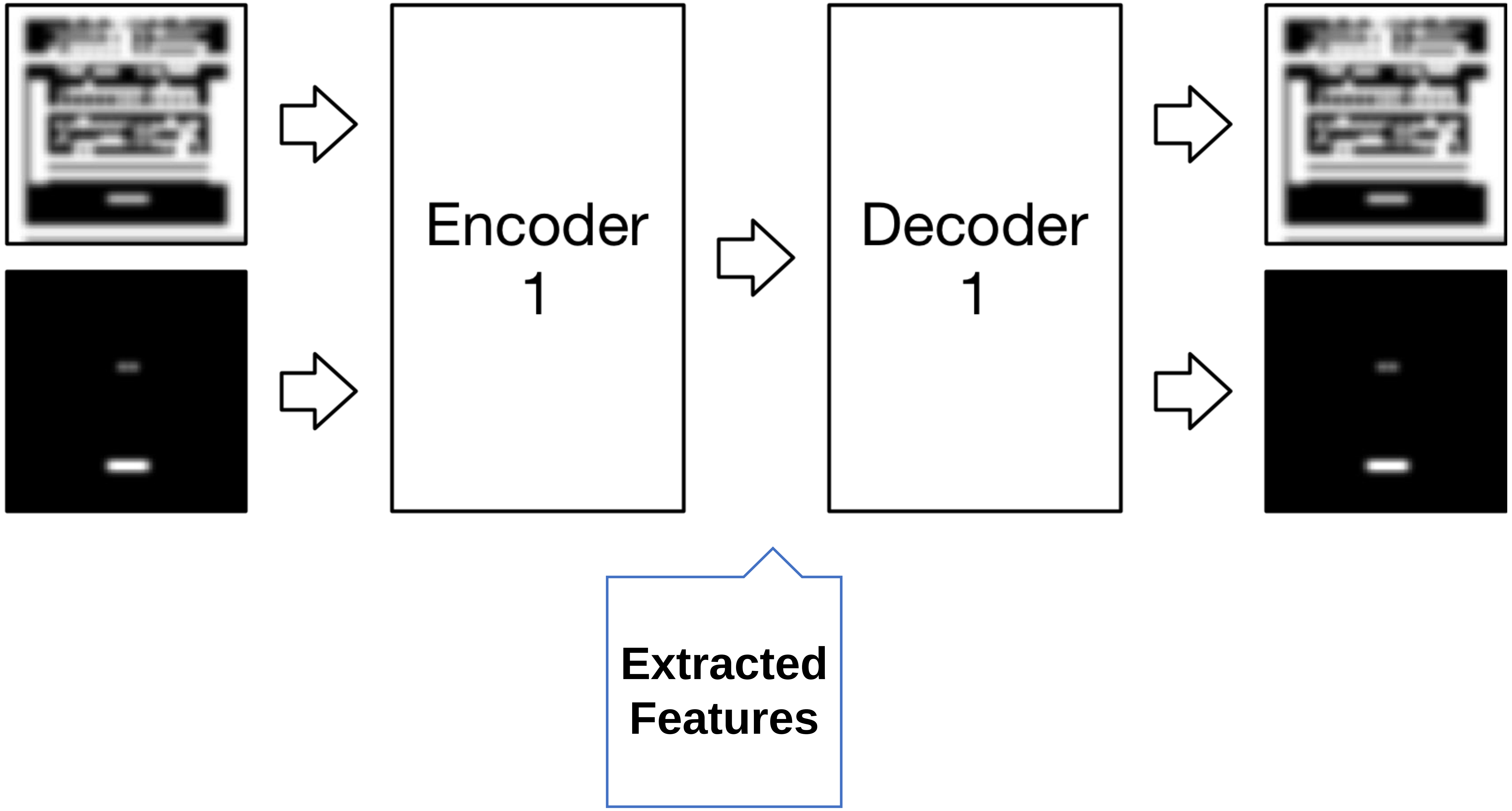
Inference Phase

3-Stage Semi-supervised Training Algorithm



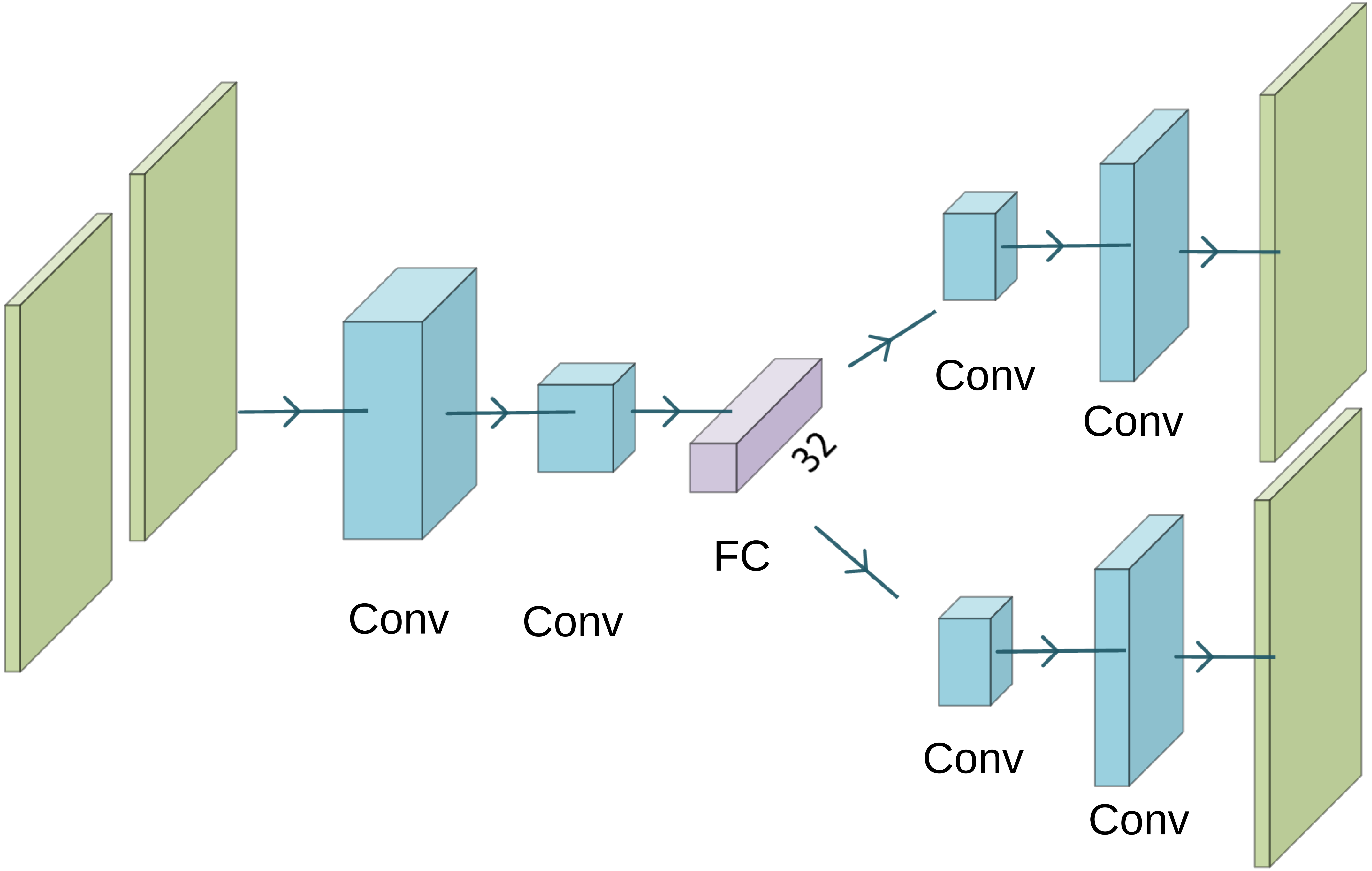
- Labeled layouts are hard to get
- Could rely on unlabeled data to help train the model

Stage 1: Unsupervised Feature Extraction using VAE

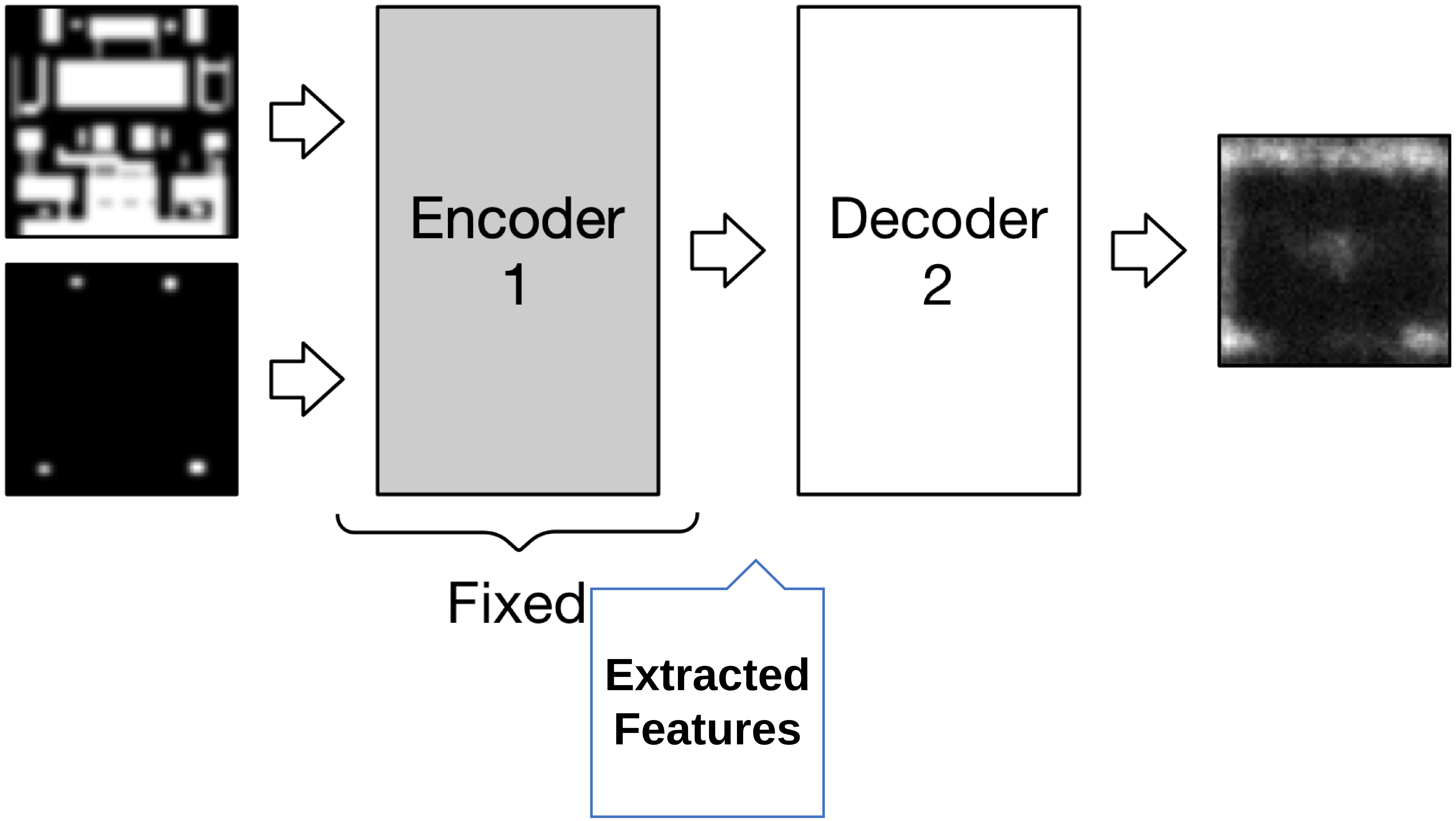


Use cheap unlabeled data to learn a general feature extraction

Network Architecture: Unsupervised for Stage 1

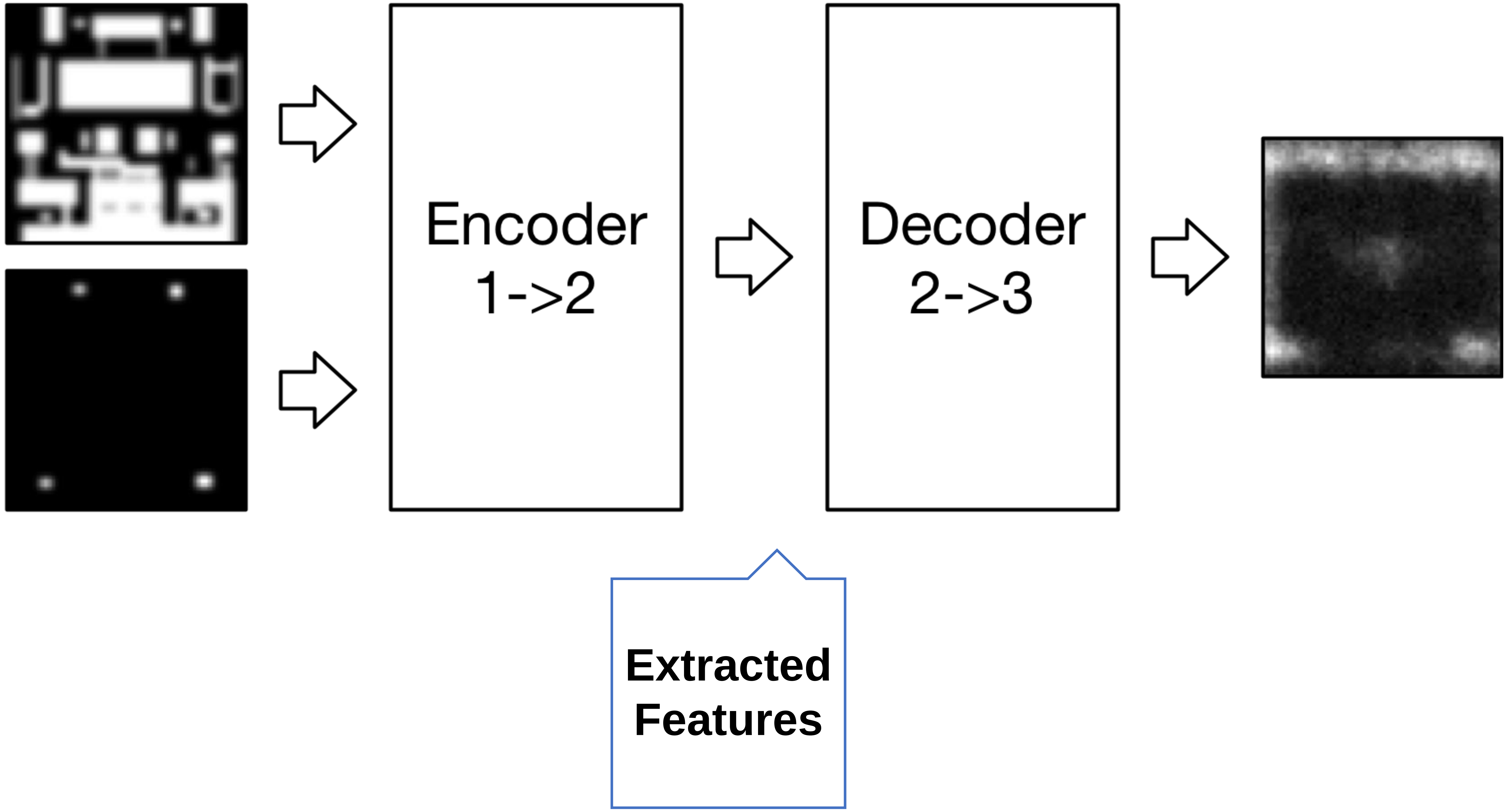


Stage 2: Supervised Decoder Training



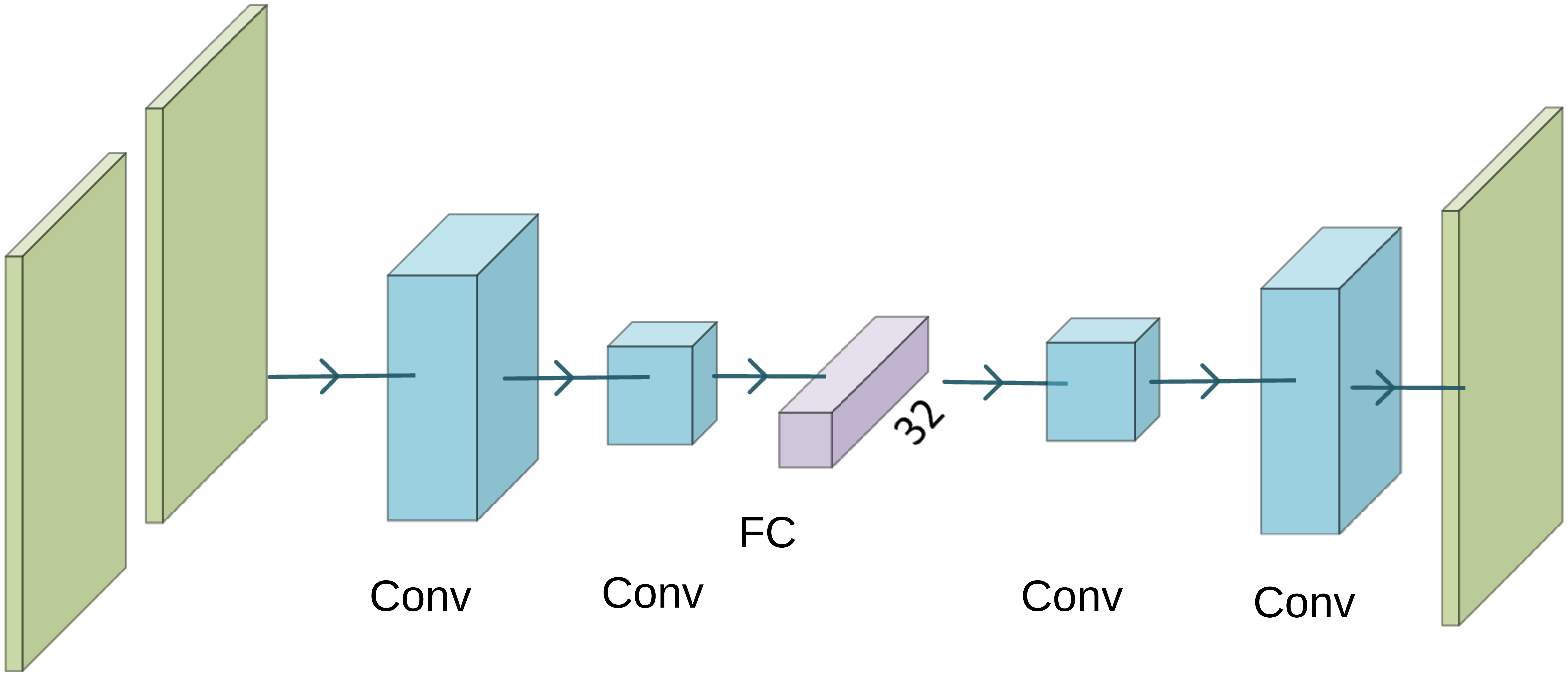
Fix the feature extraction to learn the generative model

Stage 3: Supervised Decoder Fine-Tune



Fine-tune the network for better accuracy with lower learning rate

Network Architecture: Supervised for Stage 2&3

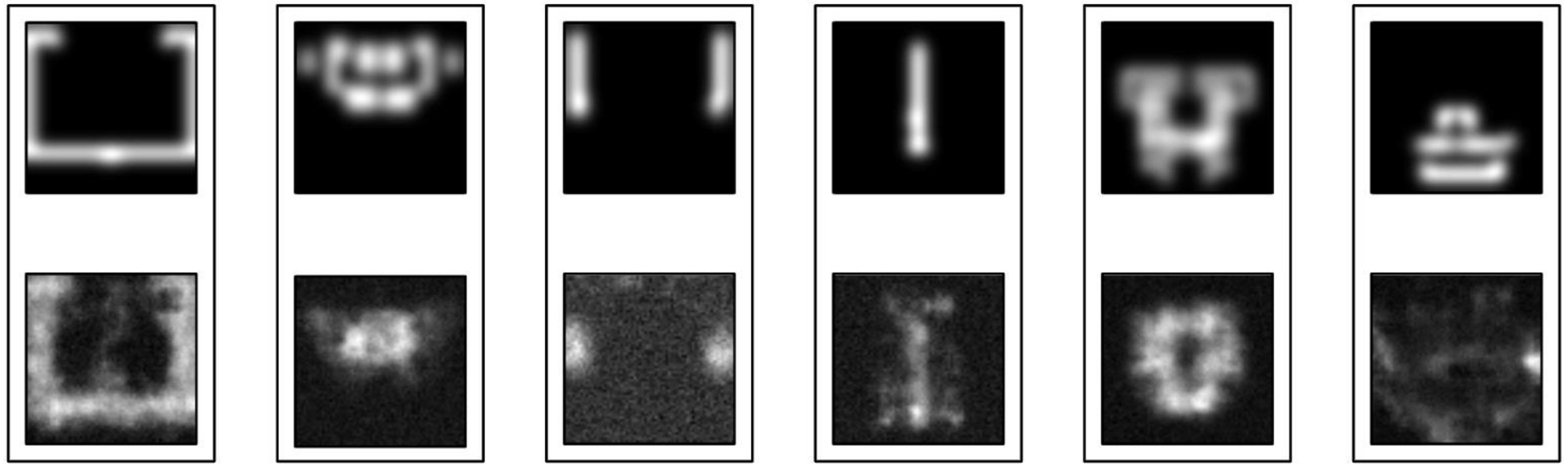


Framework Implementation and Environment Setup

- Data preprocessing: C++
- ML model: Python with Tensorflow
- Router: Modified maze routing in C++
- Simulation: Cadence ADE simulator with TSMC 40nm PDK

Experimental Result Examples

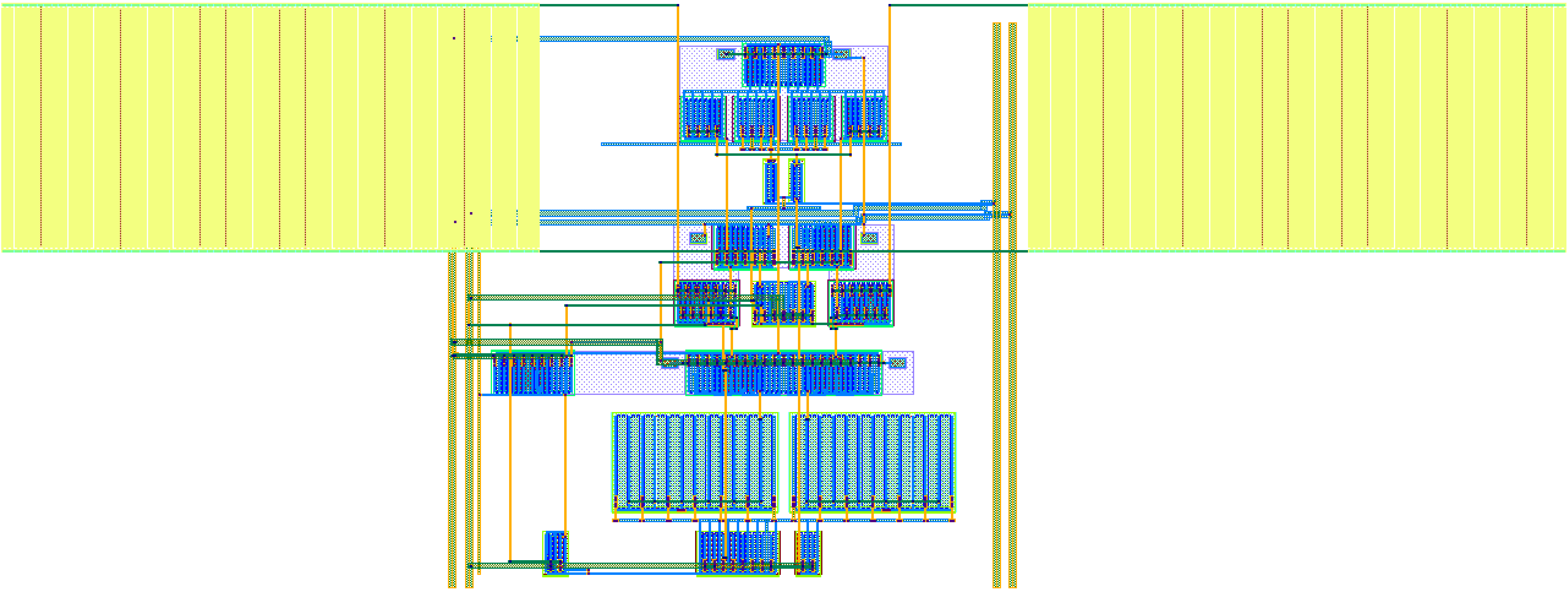
Model Output



Ground Truth

Prediction

Routed Layout



Experimental Results: Simulation Results

- Test on comparators and OTAs
- Evaluate with post layout simulation
- Compare with manual layout and previous methods

COMP1	Schematic	Manual	w/o guide	GeniusRoute
Offset (uV)	/	480	2530	830
Delay (ps)	102	170	164	163
Noise (uVrms)	439.8	406.6	439.7	420.7
Power (uW)	13.45	16.98	16.82	16.8

Closer results to the manual layout

Experimental Results: More Simulation Results

COMP1	Schematic	Manual	w/o guide	GeniusRoute
Offset (uV)	/	480	2530	830
Delay (ps)	102	170	164	163
Noise (uVrms)	439.8	406.6	439.7	420.7
Power (uW)	13.45	16.98	16.82	16.8

COMP2	Schematic	Manual	w/o guide	GeniusRoute
Offset (uV)	/	550	1180	280
Delay (ps)	102	196	235	241
Noise (uVrms)	439.8	380.0	369.6	367.8
Power (uW)	13.45	20.28	20.23	20.15

OTA	Schematic	Manual	wo/ guide	GeniusRoute
Gain (dB)	38.20	37.47	36.61	37.36
PM (degree)	64.66	72.46	94.68	76.40
Noise (uVrms)	222.0	223.7	292.7	224.8
Offset (mV)	/	0.88	3.21	0.39
CMRR (dB)	/	59.61	58.52	59.15
BW (MHz)	110.5	102.5	232.1	107.3
Power (uW)	776.93	757.35	715.11	787.82

Conclusion

GeniusRoute

- A new methodology to automatic learn from human layout and apply in automatic flow
- Semi-supervised learning algorithm for data-efficiency
- Experimental results show closed-to-human post layout simulation

Future directions

- How to overcome the challenge of obtaining human layouts for labeled data

Thank you!

Backup

TABLE I: Network configurations.

Stage 1 Configuration	
Input ($2 \times 64 \times 64$ image)	
conv/$5 \times 5 \times 64$	
conv/$5 \times 5 \times 128$	
FC/64	
Latent Variables (32)	
FC/$16 \times 16 \times 64$	FC/$16 \times 16 \times 64$
deconv/$4 \times 4 \times 32$	deconv/$4 \times 4 \times 32$
deconv/$4 \times 4 \times 1$	deconv/$4 \times 4 \times 1$
Output ($2 \times 64 \times 64$ image)	

Stage 2 & 3 Configuration
Input ($2 \times 64 \times 64$ image)
conv/$5 \times 5 \times 64$
conv/$5 \times 5 \times 128$
FC/64
Latent Variables (32)
FC/$16 \times 16 \times 64$
deconv/$4 \times 4 \times 32$
deconv/$4 \times 4 \times 1$
Output ($1 \times 64 \times 64$ image)

Backup

TABLE II: Runtime. (seconds)

	W/o guide	This work
COMP1	11.7	26.2
COMP2	2.3	21.3
OTA	49.6	55.4

TABLE III: Runtime breakdown. (seconds)

	PP	MI	DR
COMP1	<1	17.6	8.6
COMP2	<1	17.6	3.7
OTA	3.5	13.2	38.7