DRC Hotspot Prediction at Sub-10nm Process Nodes Using Customized Convolutional Network

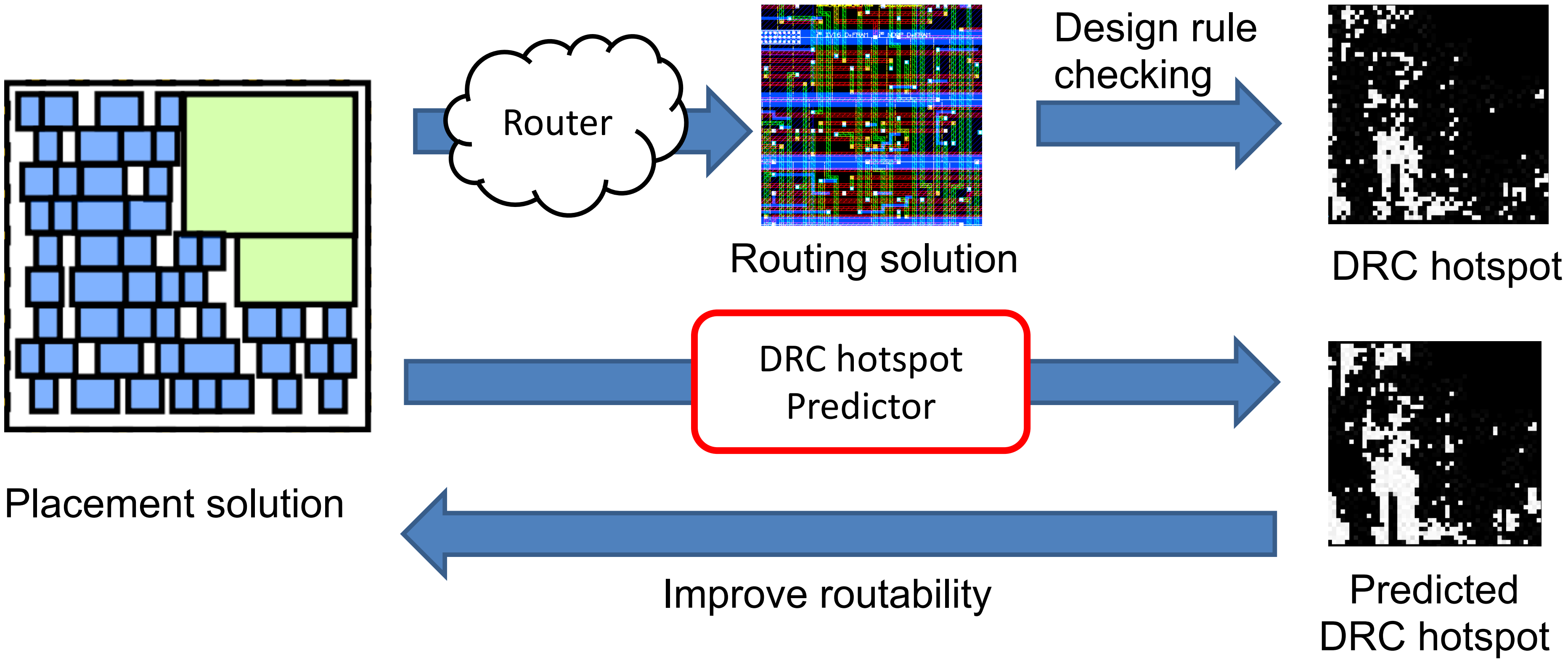
Rongjian Liang[1], Hua Xiang[2], Diwesh Pandey[2], Lakshmi Reddy[2], Shyam Ramji[2], Gi-Joon Nam[2], Jiang Hu[1]

1 Department of Electrical & Computer Engineering, Texas A&M University
2 IBM Research

Outline

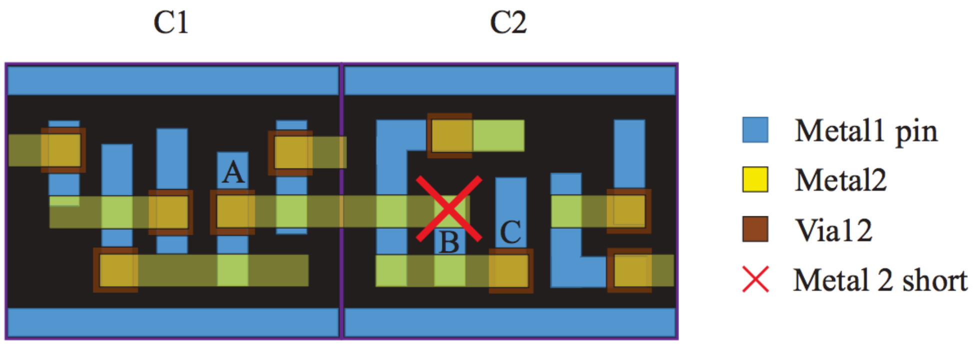
- Introduction
- Previous Works
- Feature Selection
- J-Net Convolutional Network Architecture
- Results
- Conclusion

DRC Hotspot Prediction in Placement



Challenges: Pin Accessibility

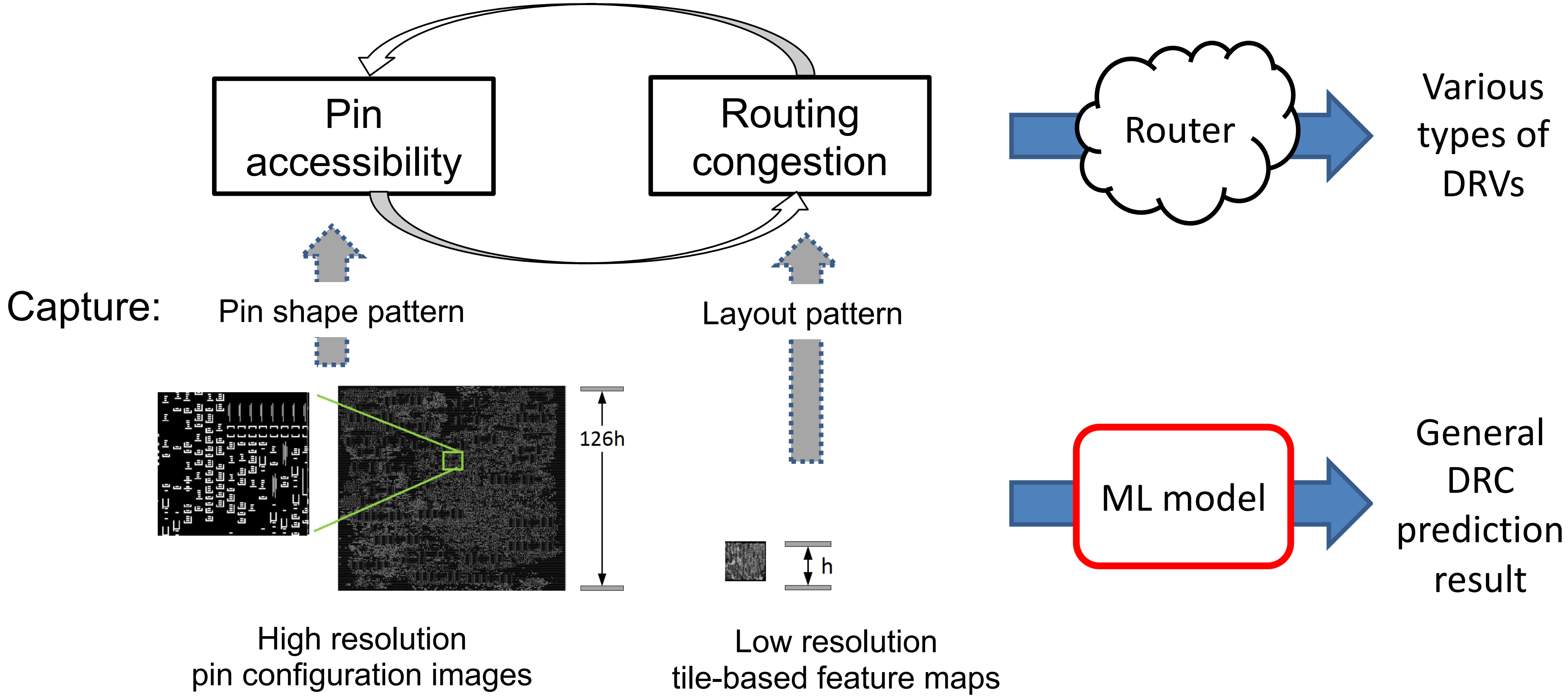
Pin accessibility is an important cause of DRVs, at *sub 10 nm nodes*.



Example of pin access problem[1]

[1] Tao-Chun Yu et al. Pin Accessibility Prediction and Optimization with Deep Learning-based Pin Pattern Recognition. DAC 2019.

Challenges: Mixed Resolution



Contributions

- A general DRC hotspot prediction technique does *not rely on global routing*
- Emphasizing both *pin accessibility* and *routing congestion*
- A customized convolutional network that address the *mixed input resolution issue*

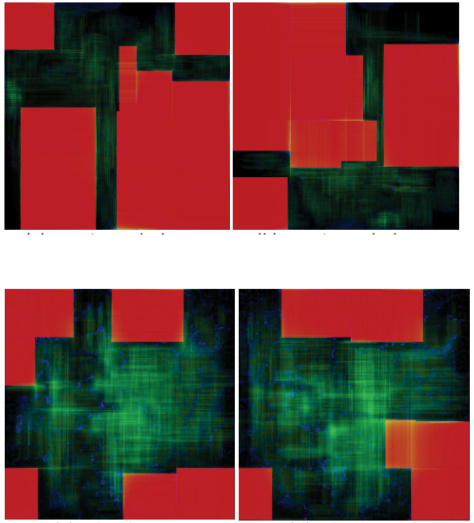
Outline

- Introduction
- Previous Works
- Feature Selection
- J-Net Convolutional Network Architecture
- Results
- Conclusion

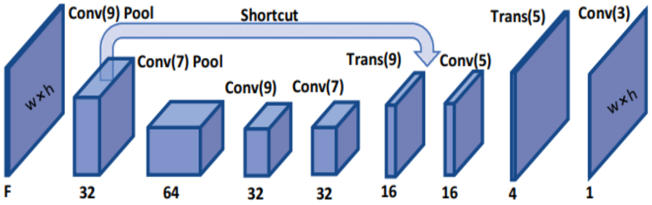
Previous Works

FCN

Zhiyao Xie et al. RouteNet: routability prediction for mixed-size designs using convolutional neural network. ICCAD 2018.



Tile-based layout feature maps



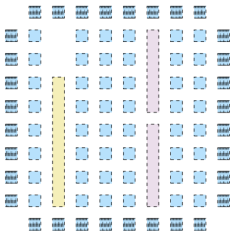
FCN Network

DRC Hotspot

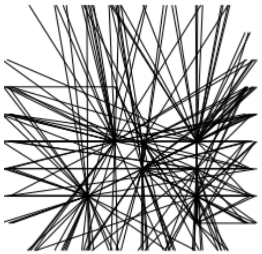
- Using global routing congestion
- Not consider pin accessibility

Previous Works

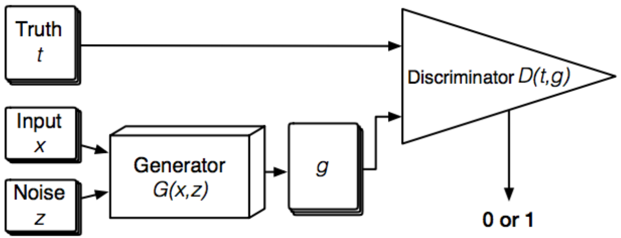
cGAN Cunxi Yu et al. Painting on Placement: Forecasting Routing Congestion using Conditional Generative Adversarial Nets. DAC 2019.



Floor plan image



Connectivity image



cGAN Network

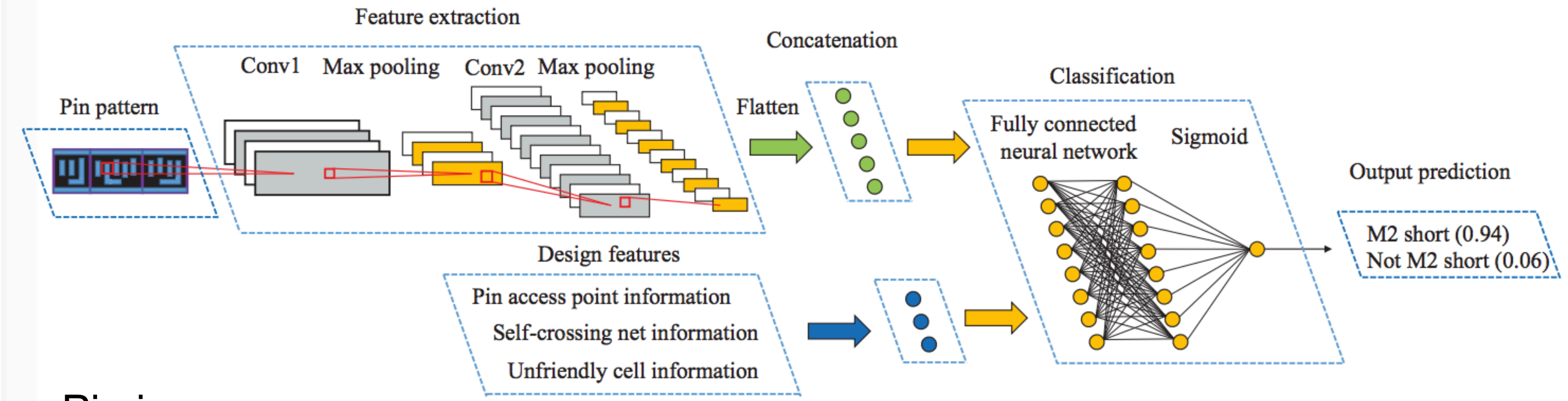
FPGA routing congestion

- Routing congestion != DRV

Previous Works

CNN

Tao-Chun Yu et al. Pin Accessibility Prediction and Optimization with Deep Learning-based Pin Pattern Recognition. DAC 2019.



Pin image covering two cells

Additional features

CNN Network

M2 short

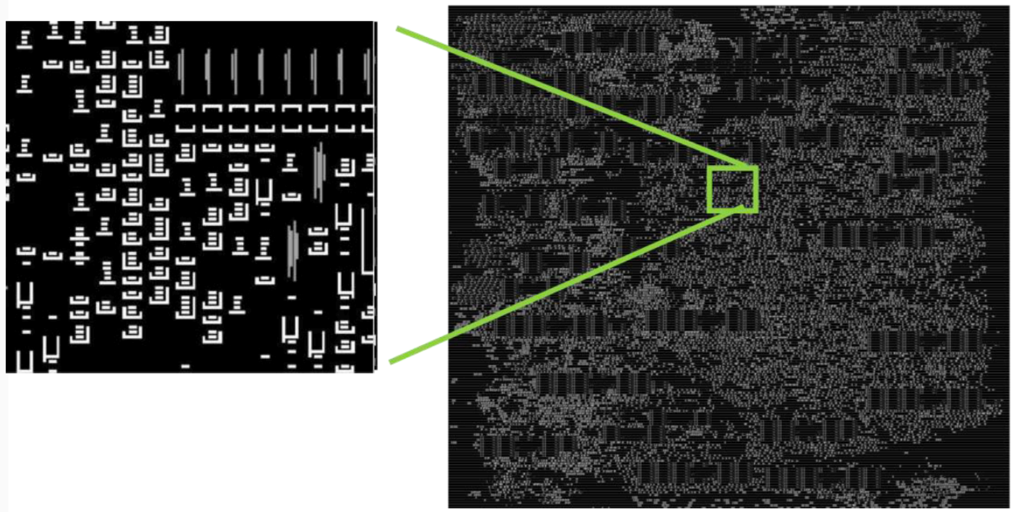
- Only M2 short
- Not consider layout information

Outline

- Introduction
- Previous Works
- Feature Selection
- J-Net Convolutional Network Architecture
- Results
- Conclusion

High Resolution Pin Configuration Image

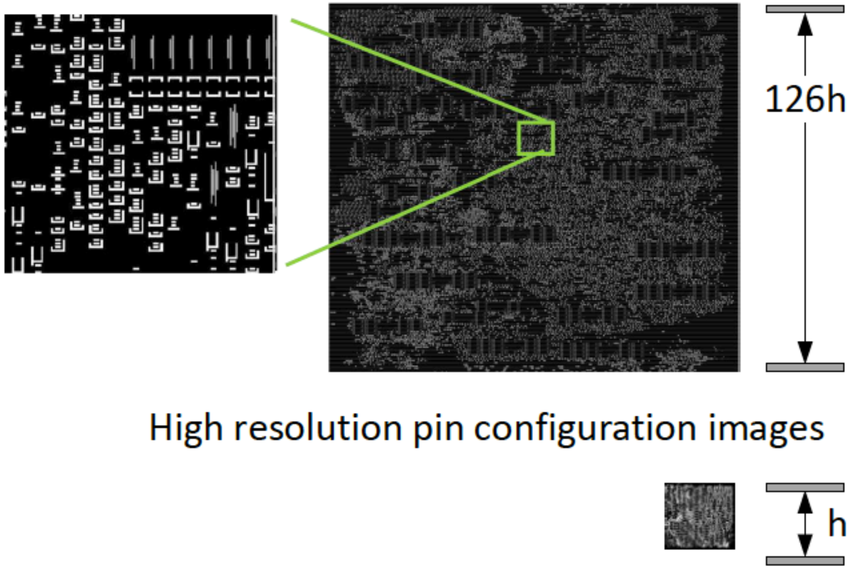
- One image for one layer where pins reside
- Resolution is high enough to show pin shape clearly
- 0 for empty space
  1 for pin access points



Pin configuration image

Low Resolution Tile-based Feature Maps

- Resolution is two orders lower than that of pin images
- Routing resource features:
  Percentage of a tile area that is occupied by IPs
- Connection features:
  #local nets and #global nets
- Each tile is 1.26μm * 1.26μm large



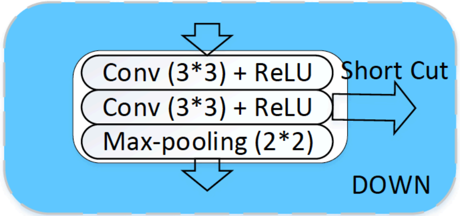
High resolution pin configuration images

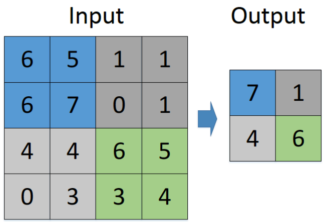
Low resolution tile-based feature maps

Outline

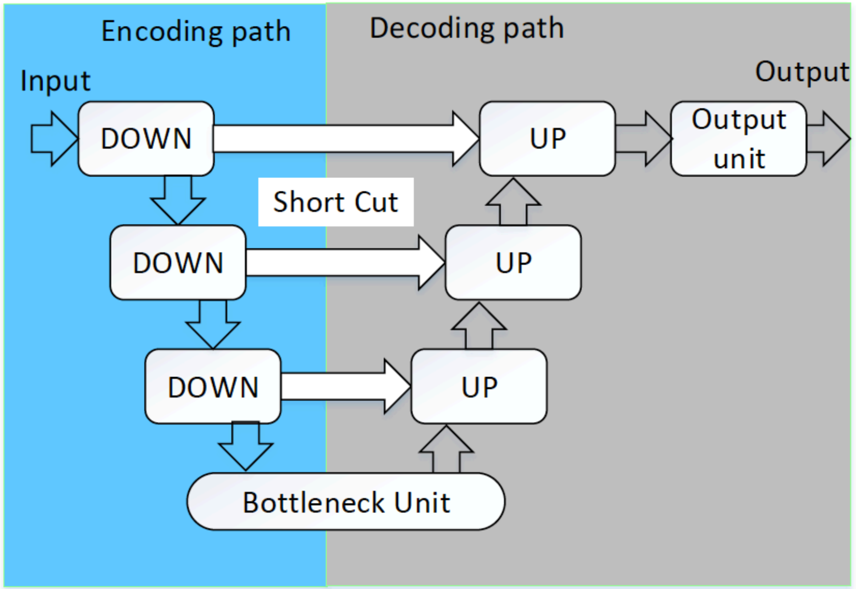
- Introduction
- Previous Works
- Feature Selection
- J-Net Convolutional Network Architecture
- Results
- Conclusion

Background on U-Net

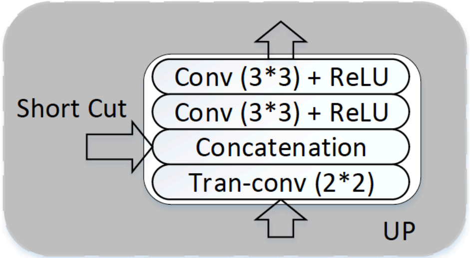


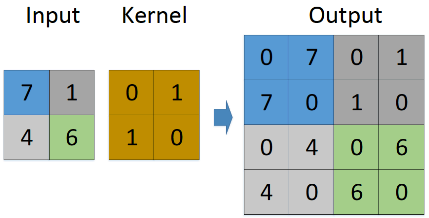


Max-pooling



Multi-level U-Net architecture

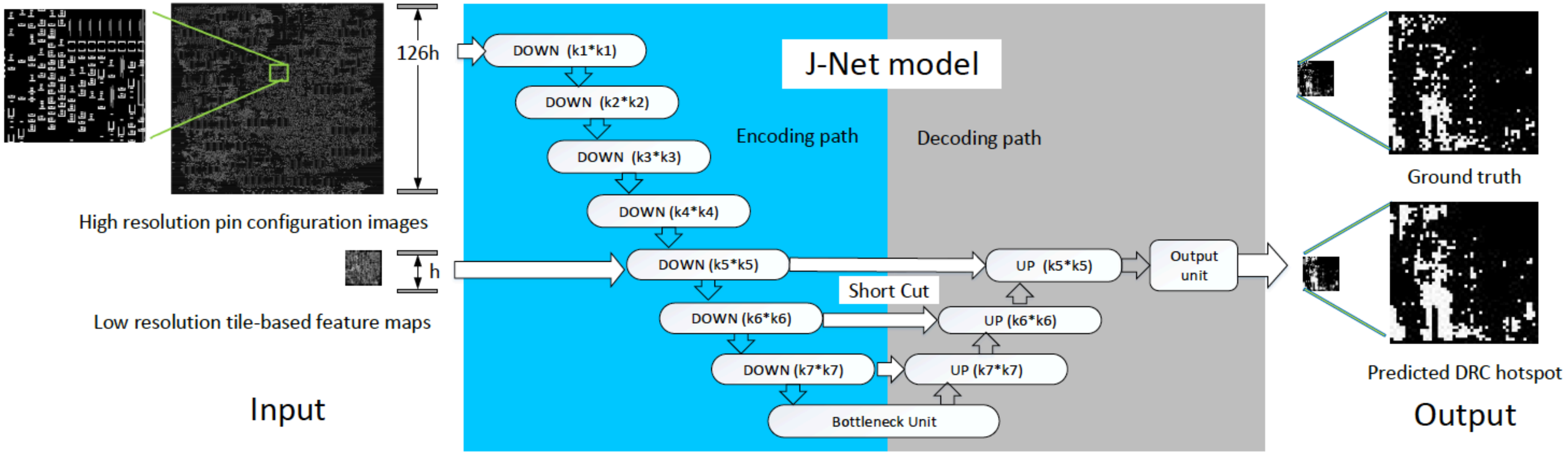




Transposed convolution

Proposed J-Net

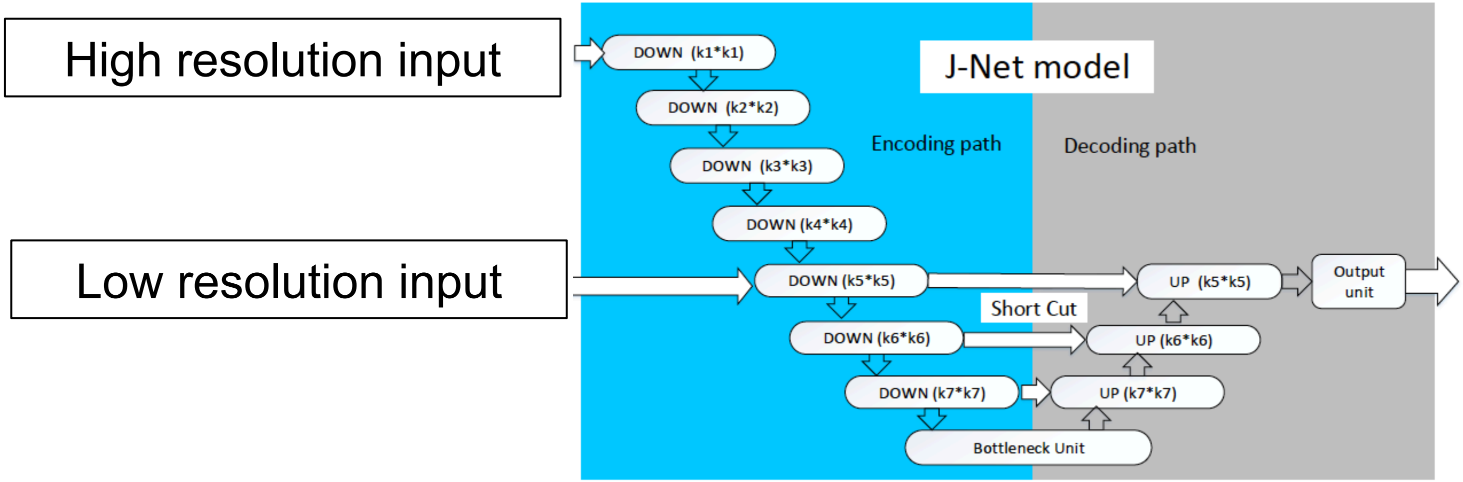
- Extension of U-Net
- Handle mixed resolution input and output



J-Net architecture

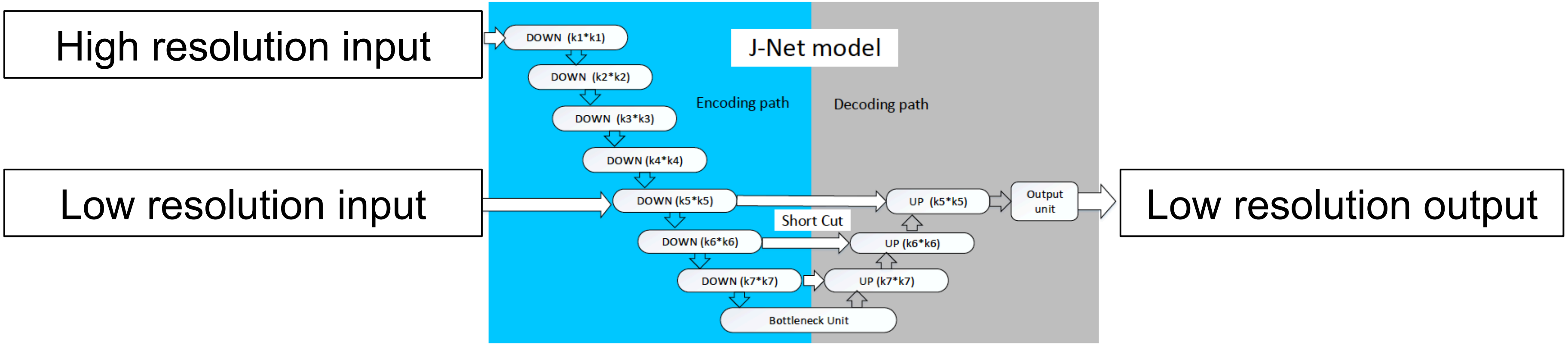
J-Net Characteristic 1

Input channels of different resolutions are fed into different levels at the encoding path



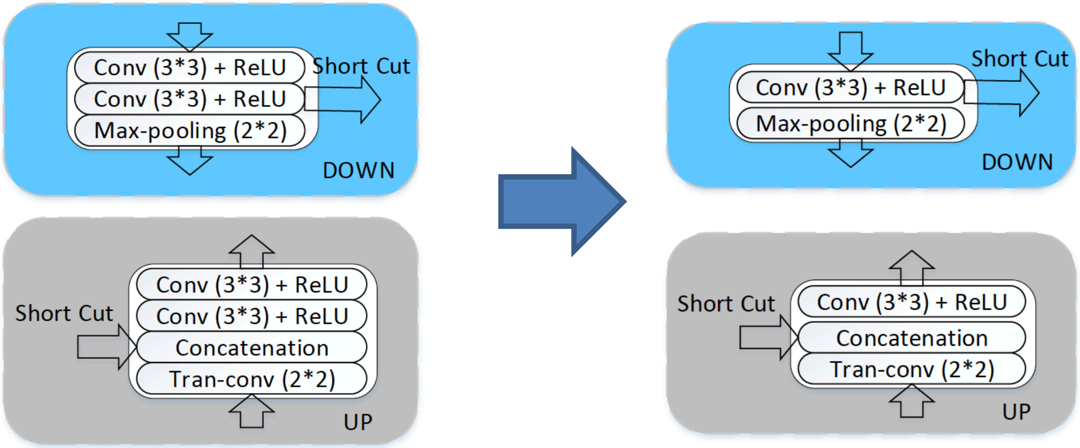
J-Net Characteristic 2

The number of decoder levels is less than that of encoder



J-Net Characteristic 3

The number of convolution operations in each down-sampling/up-sampling unit is reduced from 2 to 1



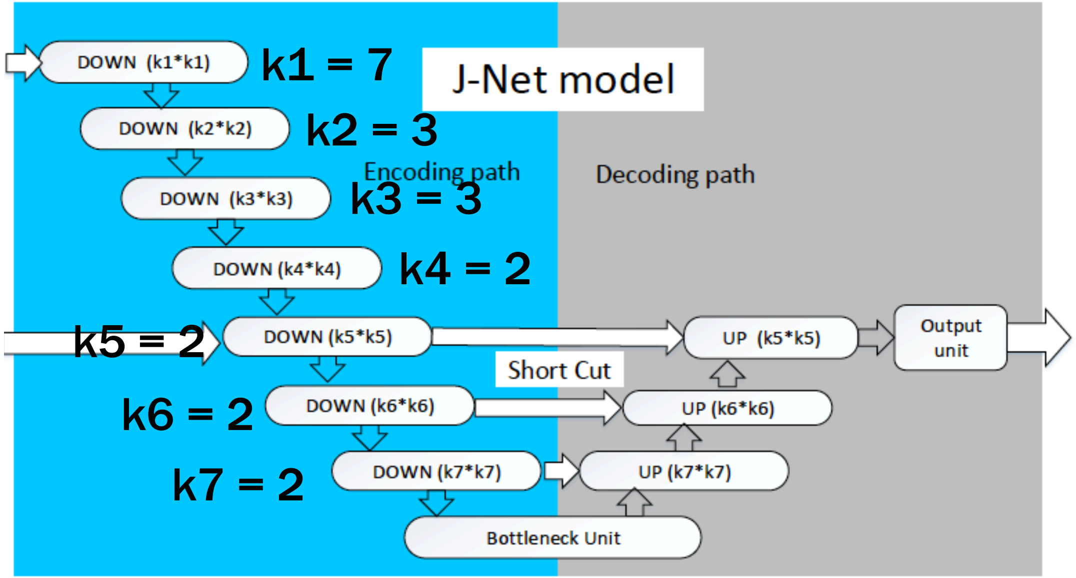
Reduce parameters -> Reduce the risk of overfitting and memory usage.

J-Net Characteristic 4

Automatic tuning of kernel size

input1 resolution:
12600*12600
(126 = 2*3*3*7)

input2 resolution:
100*100



Outline

- Introduction
- Previous Works
- Feature Selection
- J-Net Convolutional Network Architecture
- Results
- Conclusion

Experiment Setup

Number of samples:

- 12 designs
- 166 placement instances

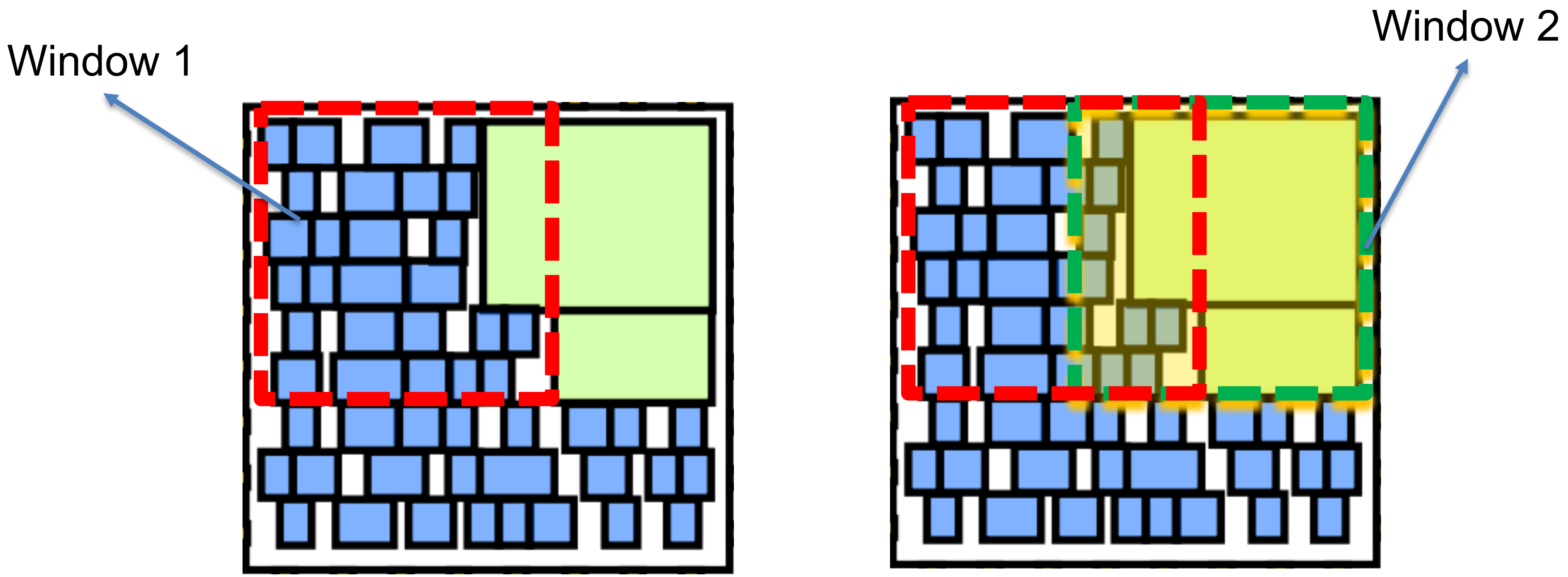
Two training & testing schemes:

- Scheme 1 : Test on *unseen placement instances*
- Scheme 2 : Test on *unseen designs*

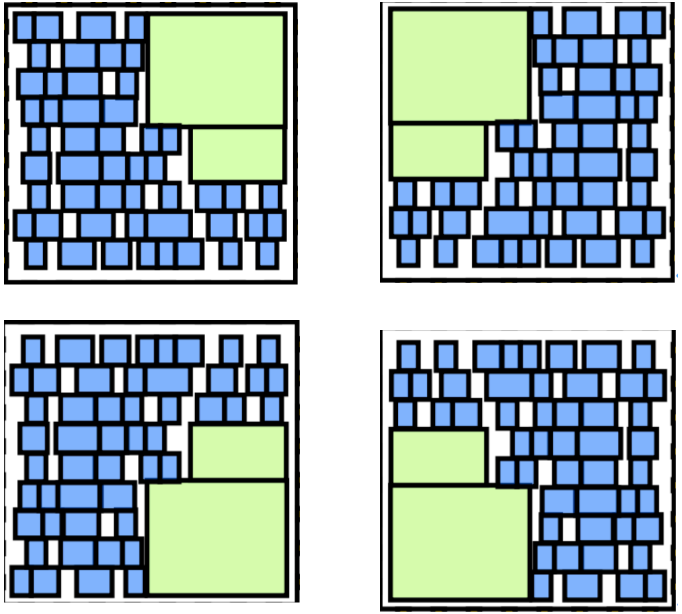
Testcase characteristics

Design	Chip Size (μm^2)	#IPs	#Gates	#Nets
D1	59.52×62.21	0	13569	15552
D2	59.52x×129.60	0	10796	17843
D3	211.20×391.39	0	328611	338846
D4	79.68×84.24	0	18283	25068
D5	122.88×233.28	0	68393	86640
D6	122.88×95.90	0	46001	49337
D7	138.24×80.35	0	35627	38456
D8	284.16×95.90	0	100566	108187
D9	76.80×401.76	1	52659	70338
D10	249.60×316.22	3	149456	191513
D11	280.32×489.89	16	81384	105678
D12	253.44×671.33	28	200454	247271

Data augmentation: Cropping

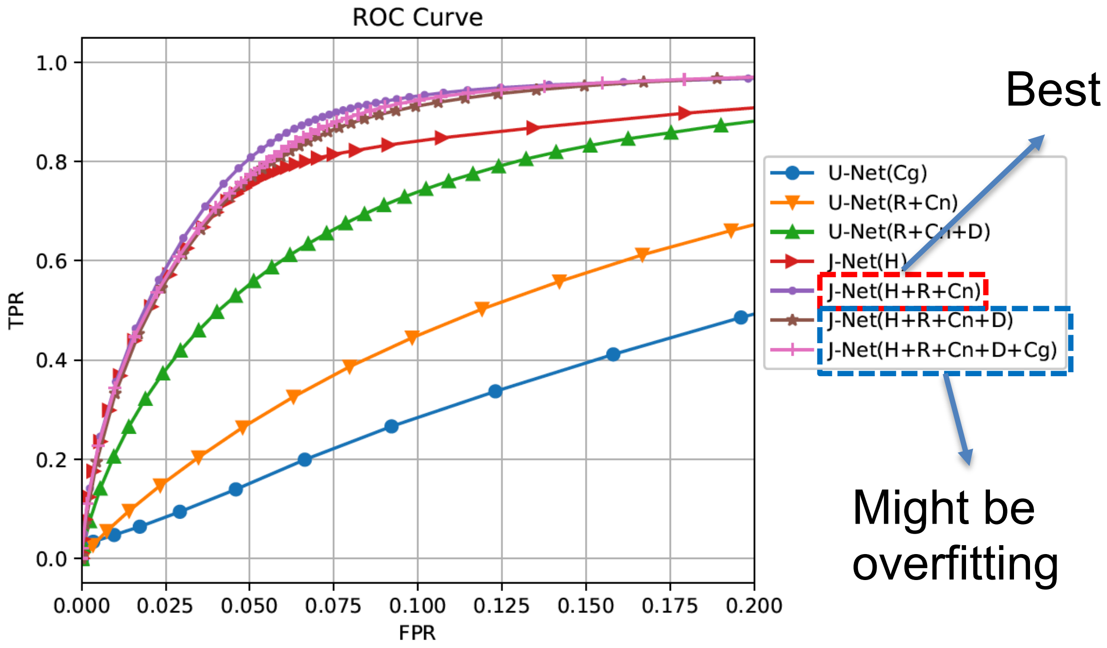


Data augmentation: Random Flipping



Scheme 1: Comparison of Features

- ROC: Receiver Operating Characteristic, tradeoff between TPR (True Positive Rate) and FPR (False Positive Rate)
- H: Pin configuration images
- R: Routing resource feature
- Cn: Connection features
- Cg: GR congestion map
- D: density features such as logic gate pin density, clock pin density, logic cell density, filler cell density, etc



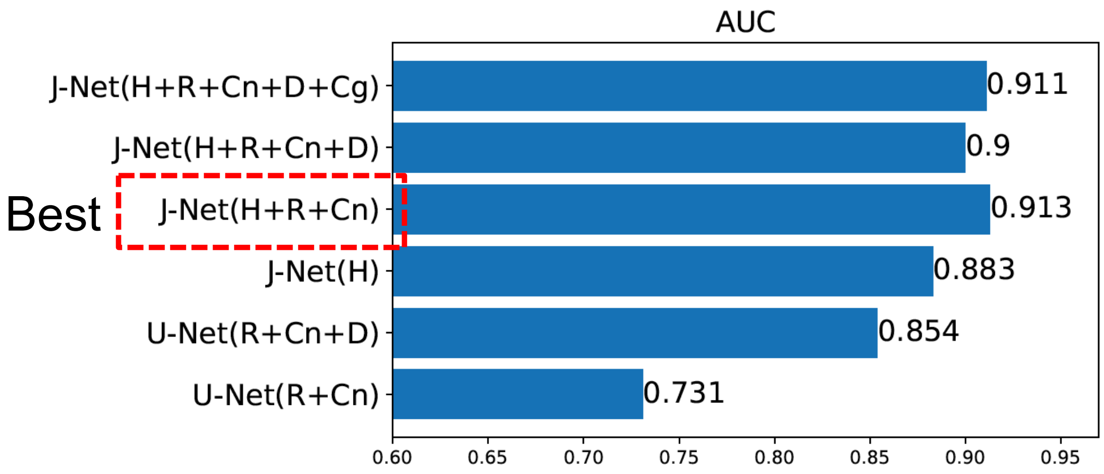
Scheme 1: Comparison of Various Methods

	Extension of previous works			Plug-in use of existing model	Customized model
Metric	**FCN**	**cGAN**	**CNN**	**U-Net**	**J-Net**
AUC of ROC	0.867	0.818	0.927	0.913	*0.958*
FPR	9.0%	9.9%	9.5%	9.6%	9.8%
TPR	56.5%	51.7%	79.2%	72.9%	*93.0%*
Precision	35.1%	31.9%	42.9%	40.6%	46.2%
F1-score	43.3%	39.5%	55.7%	52.2%	61.8%
Global routing?	Y	N	N	N	N

AUC: Area Under Curve (ideally 1.0) Precision = TP/(FP + TP) F1 = 2TP/(2TP+FP+FN)

Scheme 2: Comparison of Features

- AUC: Area Under Curve of Receiver Operating Characteristic, ideally 1.0
- H: Pin configuration images
- R: Routing resource feature
- Cn: Connection features
- Cg: GR congestion map
- D: density features such as logic gate pin density, clock pin density, logic cell density, filler cell density, etc



Scheme 2 : Comparison of Various Methods

	Extension of previous works			Plug-in use of existing model	Customized model
Metric	**FCN**	**cGAN**	**CNN**	**U-Net**	**J-Net**
AUC of ROC	0.788	0.714	0.871	0.854	*0.913*
FPR	9.1%	9.7%	9.4%	9.47%	8.90%
TPR	41.0%	38.1%	71.4%	56.1%	*78.5%*
Precision	31.3%	29.9%	43.9%	35.8%	46.2%
F1-score	32.3%	29.9%	49.4%	39.3%	54.0%

AUC: Area Under Curve (ideally 1.0) Precision = TP/(FP + TP) F1 = 2TP/(2TP+FP+FN)

Runtime

- Global routing: *several hours* for one layout design
- J-Net Training: *~ 27 hours* , can be reused across different designs
- J-Net Inference: *< 1 minute* for one layout design

Outline

- Introduction
- Previous Works
- Feature Selection
- J-Net Convolutional Network Architecture
- Results
- Conclusion

Conclusion

- A general DRC hotspot prediction technique that does *not rely on global routing*
- A customized convolutional network that address the *mixed resolution issue*
- Above 7% higher TPR, at the same FPR, than extensions of previous works

Thank you!