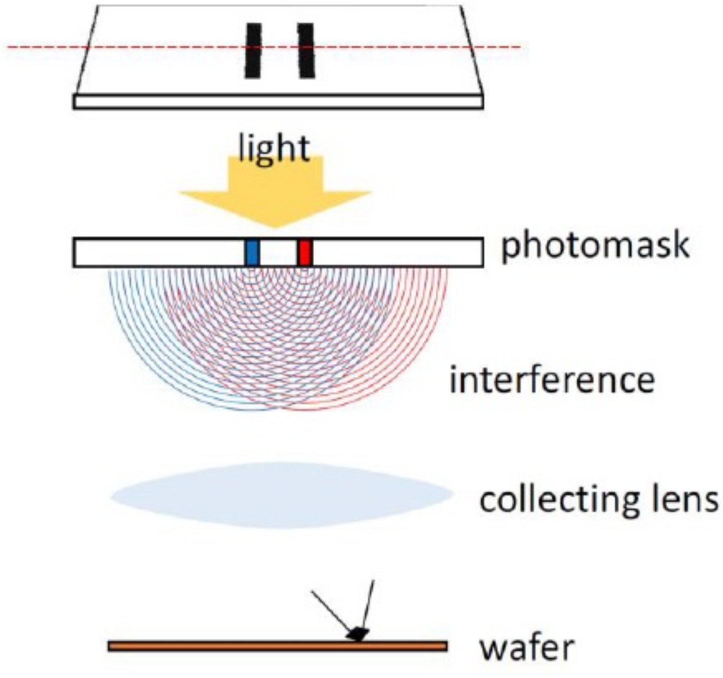
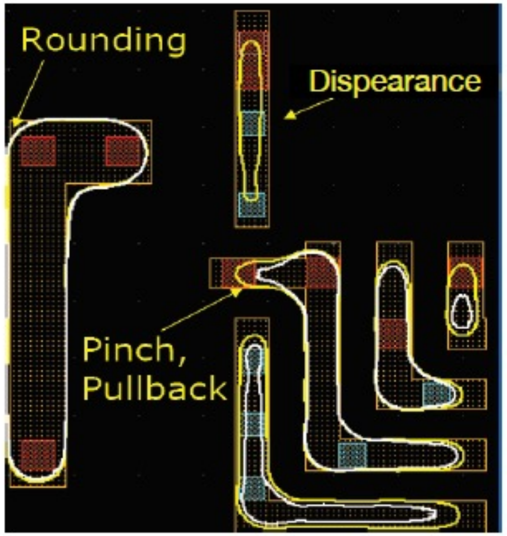
Lithography Hotspot Detection

Lithography

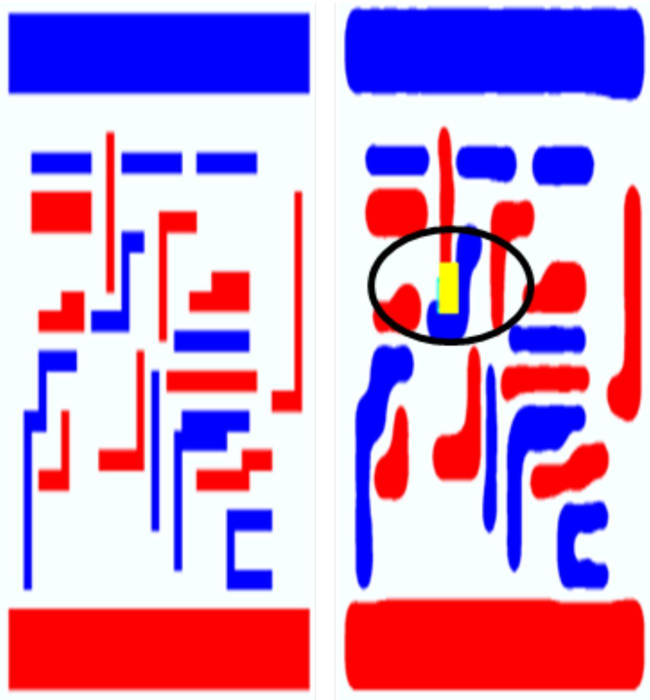
- With light diffraction through the mask
  - What you see is not what you get
  - May cause shorts/opens, performance degradation, yield loss



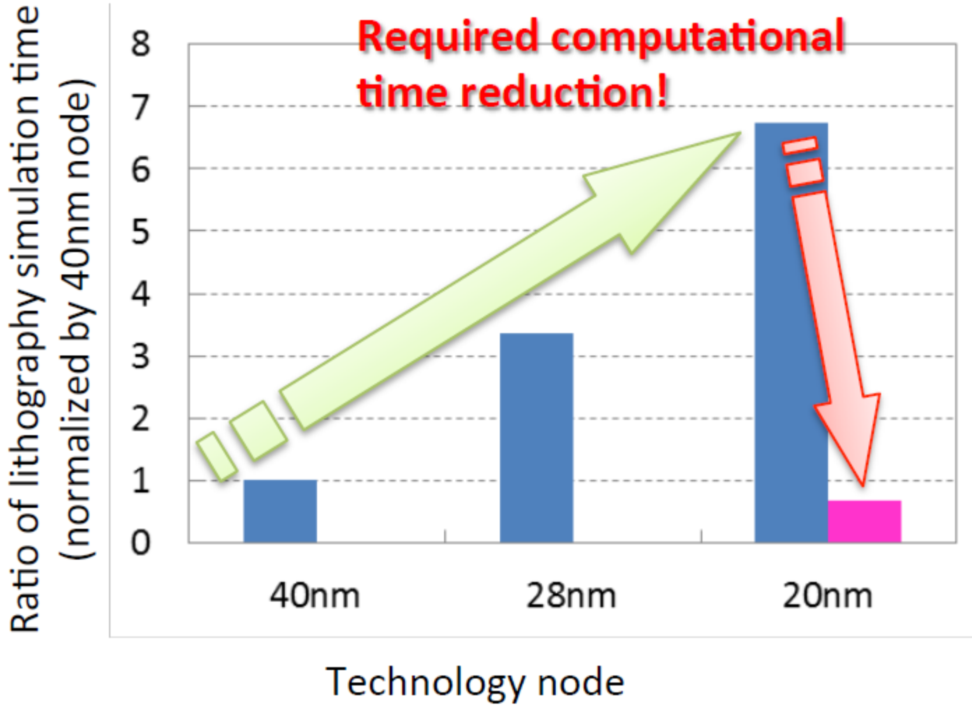


Hotspot Detection

- Hotspots: problematic layout patterns
- Hotspot detection by simulation is extremely CPU intensive

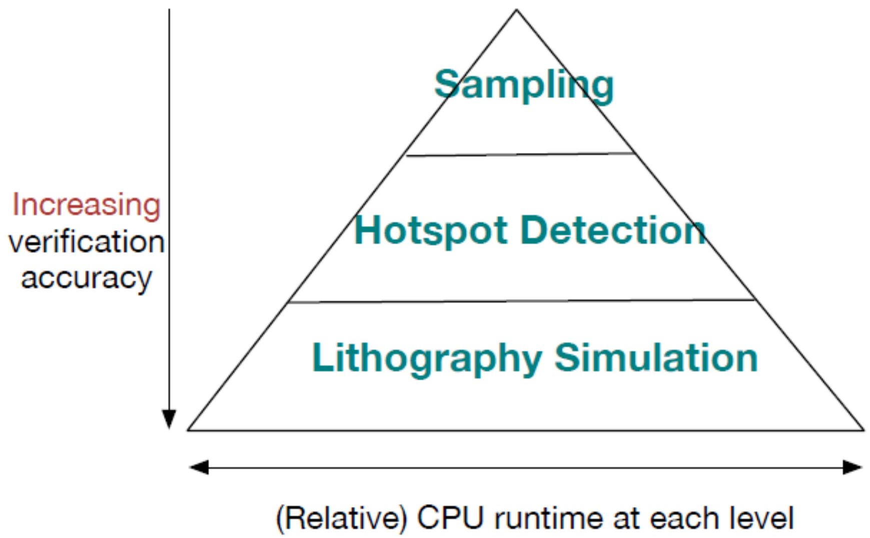


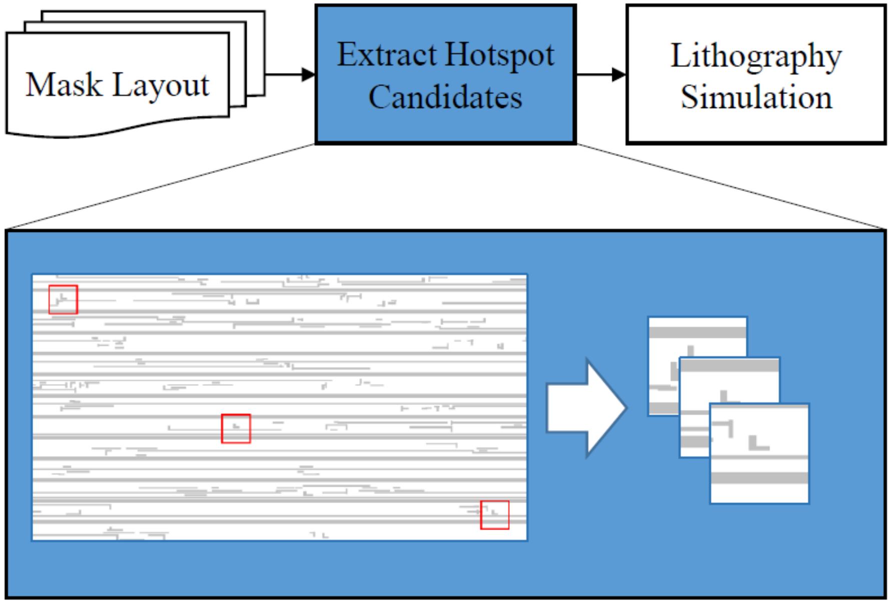
bridging hotspot



Layout Verification Hierarchy

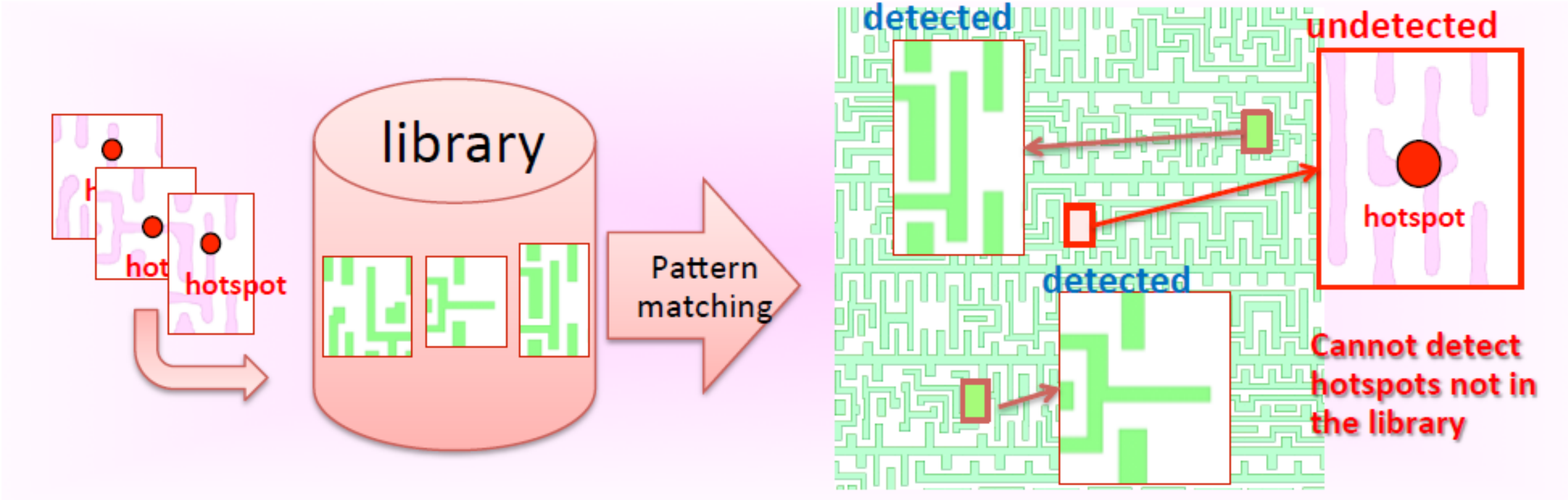
- How to avoid full-chip simulation?





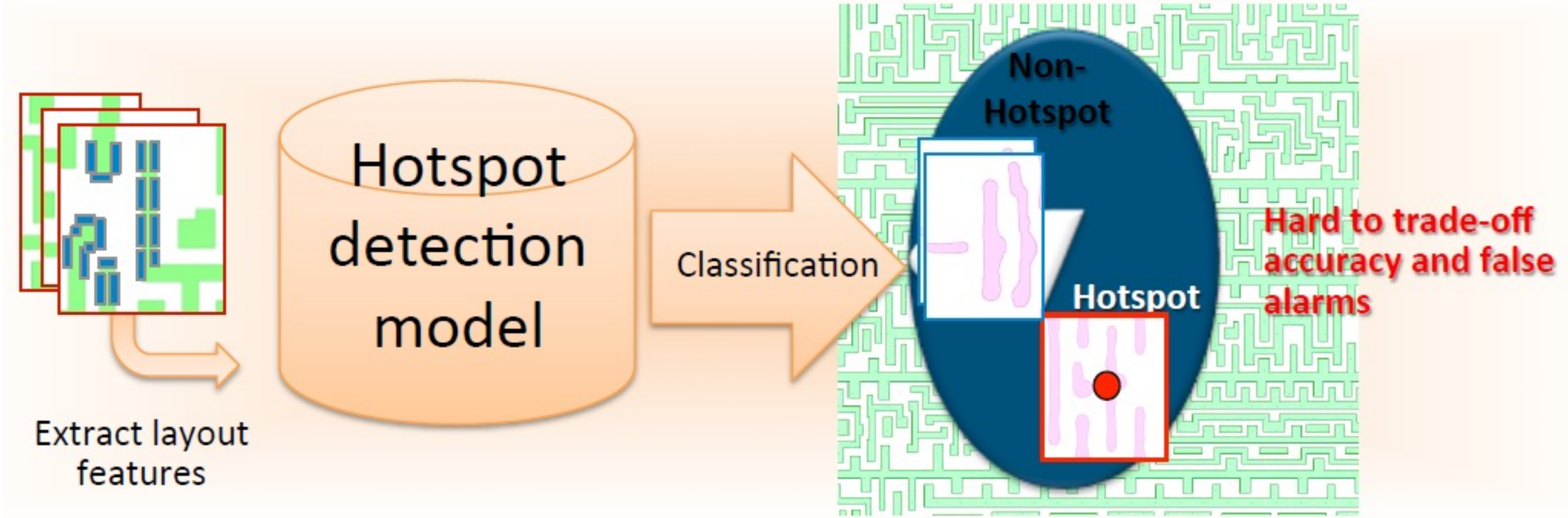
- Sampling: use a moving window to extract possibly problematic layout clips e.g. based on simple rules like polygon density, etc.
- Hotspot detection: apply coarse-grained hotspot detector
- Lithography simulation: verify reported hotspots

Pattern Matching Based Hotspot Detection



- Library keeps a set of pre-characterized hotspot patterns
- Fast and reasonably accurate
- Can't identify previously unknown hotspot patterns
- E.g. [Yu+ DAC12]

Machine Learning Based Hotspot Detection

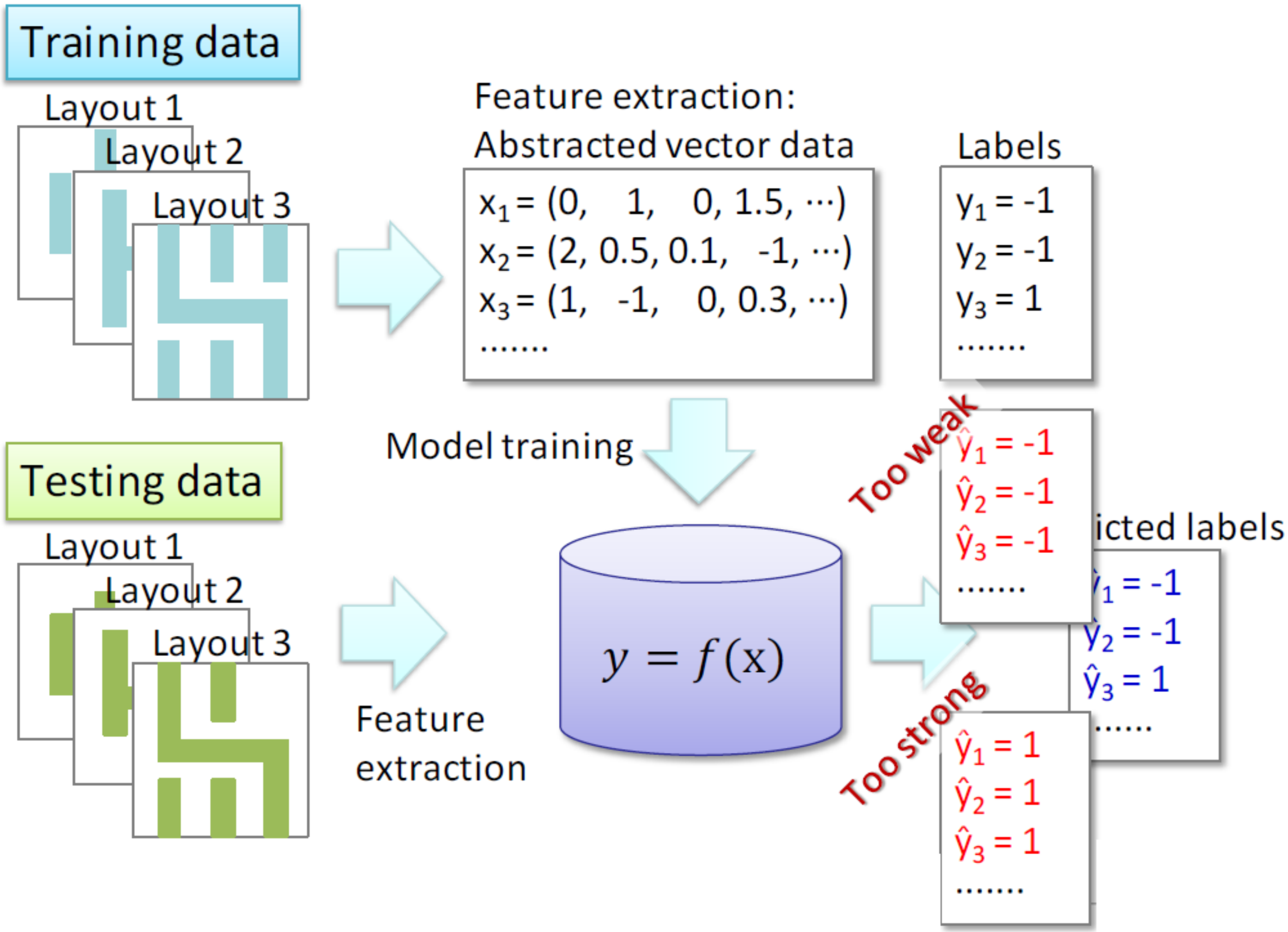


- Can predict even unseen hotspot patterns
- Generate false alarms which are not real hotspots
- Trade off between accuracy and false alarm
- E.g. [Yu+ DAC13]

Performance Metrics

- **Accuracy**: The rate of correctly predicted hotspots among the set of actual hotspots
- **False Alarm**: Number of non-hotspots reported as hotspots

Machine Learning Based Hotspot Detection



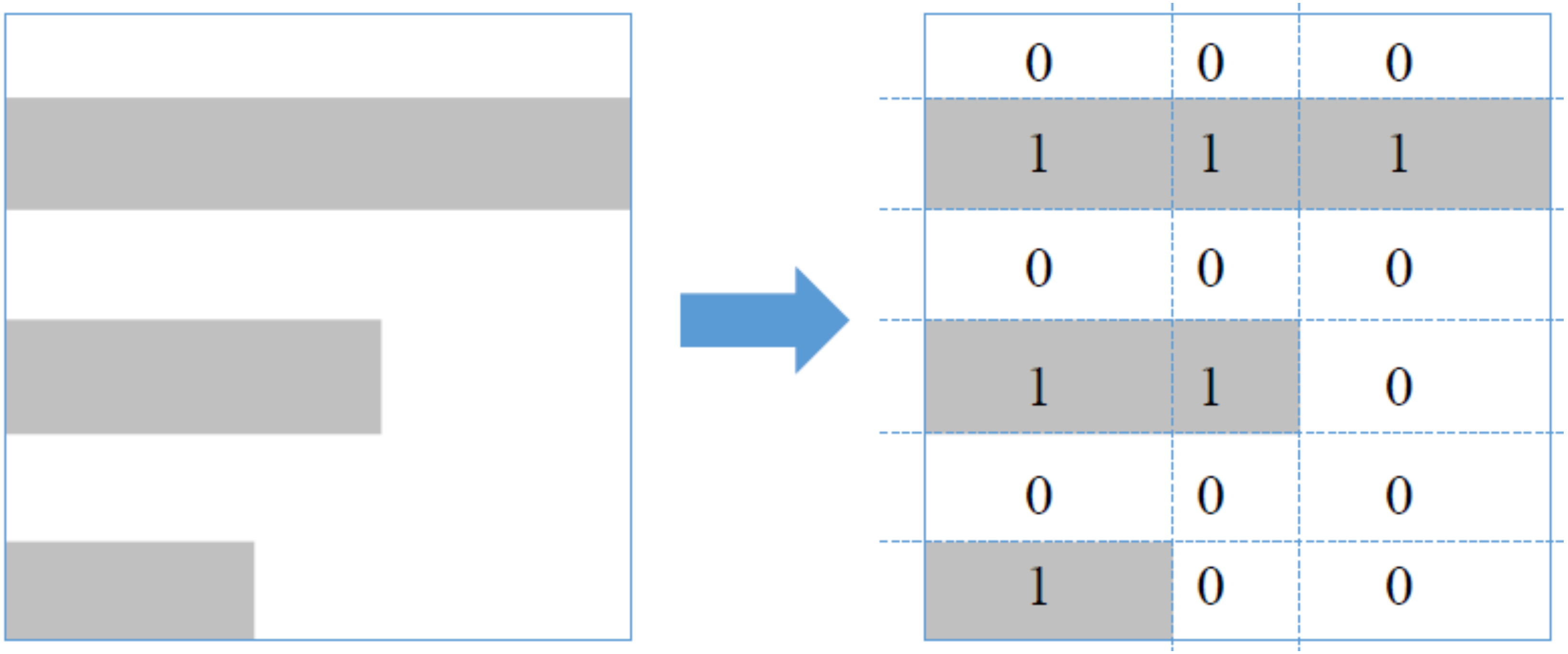
x_i and y_i are feature and label (-1 or 1 for two-class classification)

Layout Feature Extraction

- How to describe a hotspot’s environment?
- Many proposed representations e.g. graph, string, vector, matrix
- Note that representation chosen affects the detection method that we can use, run time, accuracy, etc.

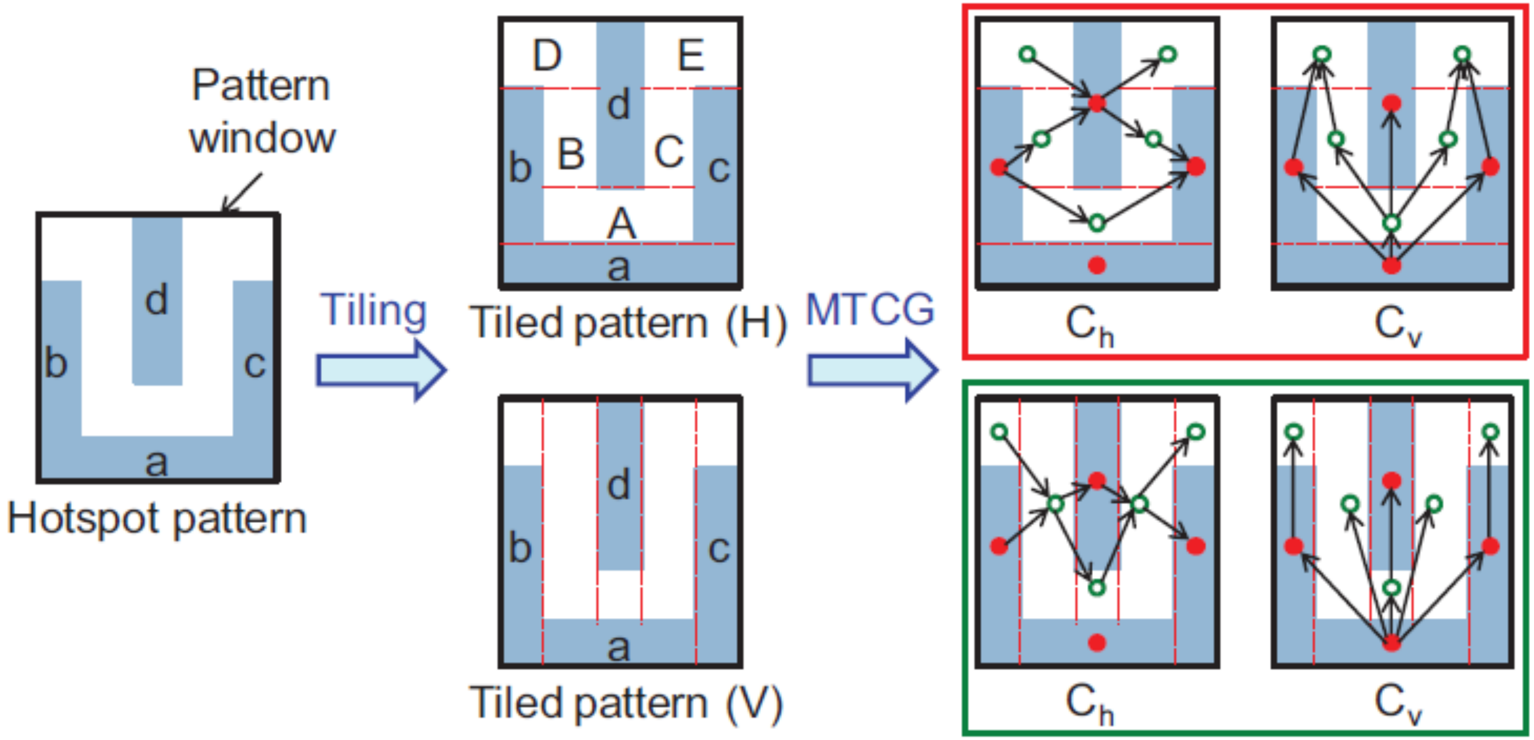
Feature Extraction by Topological Representation

- e.g. A topology matrix capturing the geometrical relationship of the patterns within a layout clip.



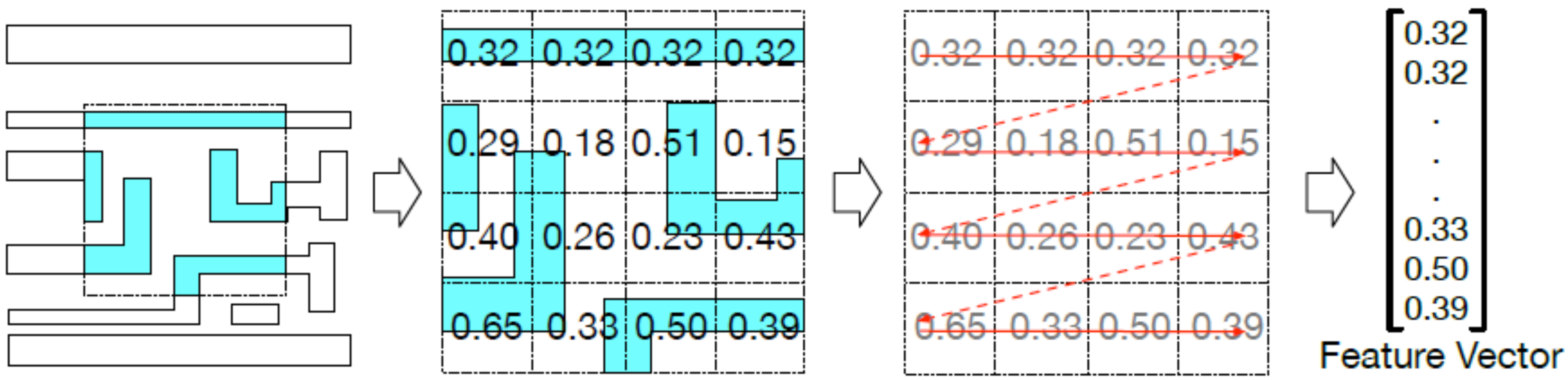
Feature Extraction by Topological Representation

- e.g. Modified Transitive Closure Graph [Yu+ DAC12]
- Use a pair of graphs C_h and C_v to capture polygon topology



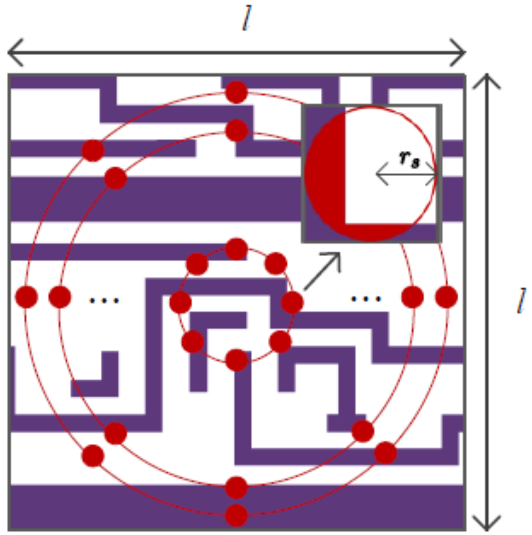
Feature Extraction by Density-Based Layout Pattern Encoding

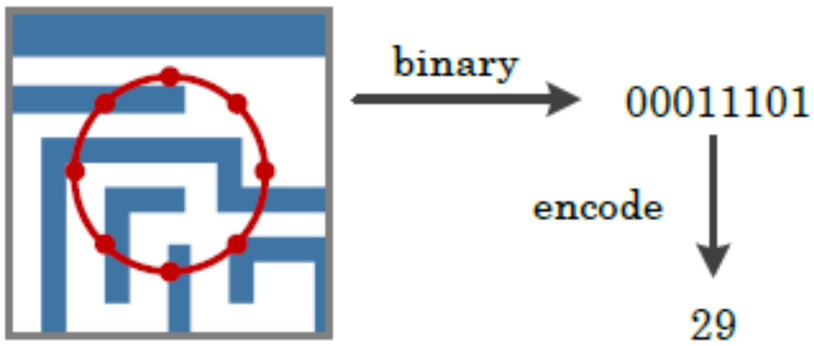
- [Wen+ TCAD14]
- Use a vector of pattern densities of imposed uniform grid



Feature Extraction by Even Concentric Circle Area Sampling

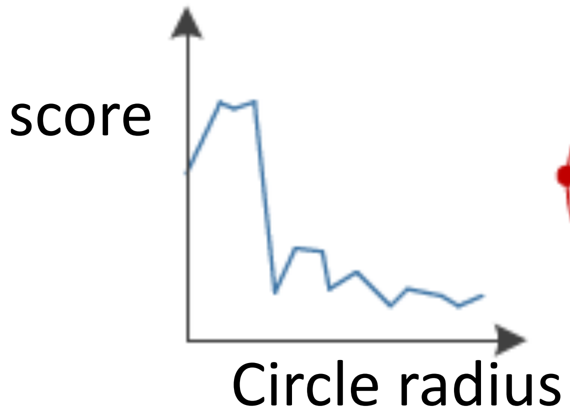
- [Zhang+ ICCAD16]
- Assume hotspot is at the center of a clip, concentric circle sampling captures the effect of light propagation
- Fixed number of sampling points on each circle, a sampling point either covers layout element (1) or not (0)
- After encoding each circle as an integer, final representation is flattened into a vector of integers

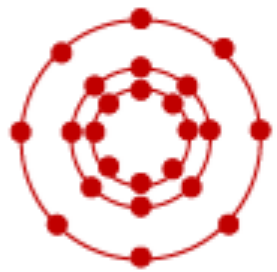




Feature Extraction by Optimized Concentric Circle Area Sampling

- [Zhang+ ICCAD16]
- Conventional concentric circle area sampling select circles evenly
- Better to select most representative circles only by dynamic programming





"score" indicates the correlation of a circle with the label variable which is layout dependent

Feature Extraction by Matrix Based Concentric Circle Area Sampling

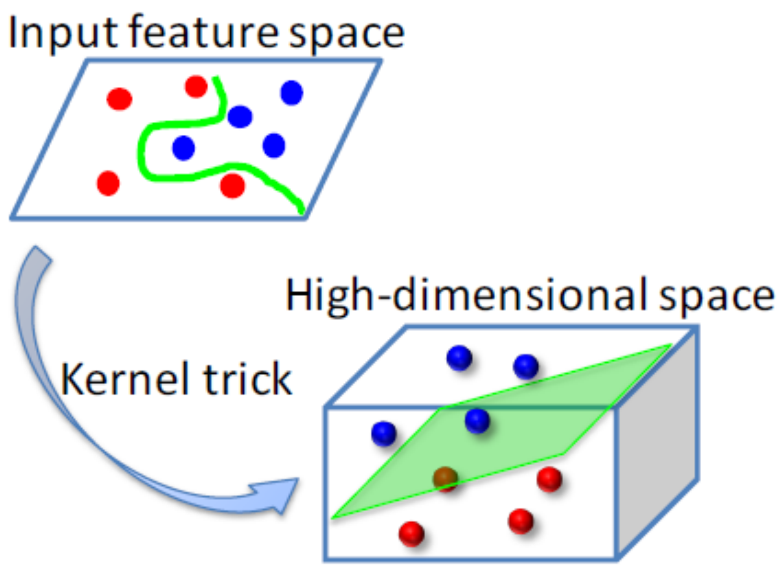
- [Zhang+ ISPD17]
- Light intensity induced by each sampling point should have proportional influence to hotspot formation, so each entry should be *0~area of sampling point* (rather than 0/1)
- Matrix-based representation
  - each concentric circle contributes a row
  - correlations among rows and columns are kept to capture effect of light interference

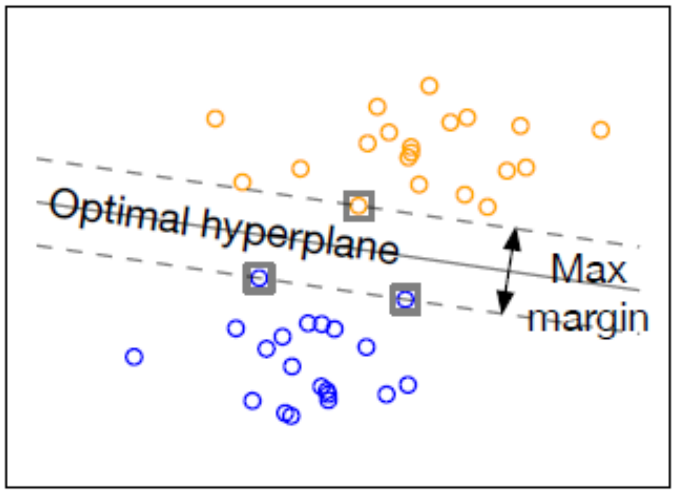
Detection Method

- How to improve accuracy and reduce false alarm?
- Many different techniques have been proposed e.g. pattern matching, machine learning algorithms (support vector machine SVM, Boosting, various neural networks, etc.), unified meta-classification, fuzzy matching, online learning

Machine Learning

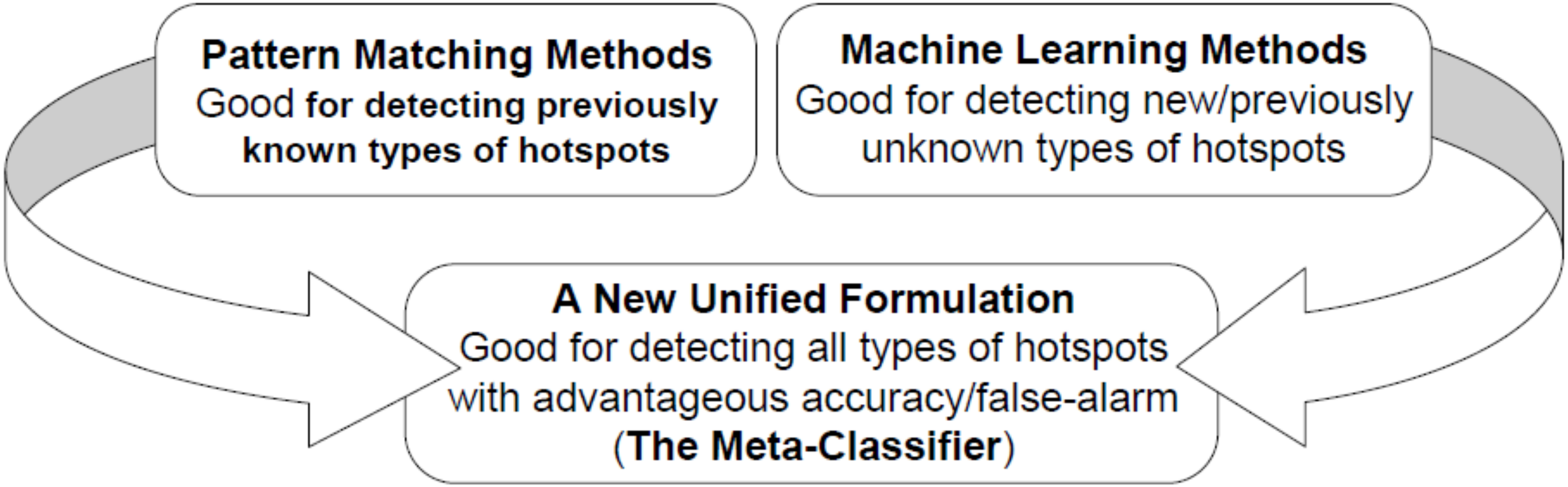
- Example. Support Vector Machine SVM
  - Given a set of training examples marked as belonging to one or the other of two classes
  - Map input space into a higher-dimensional space using a kernel function so a hyperplane can be used to separate two classes
  - Choose hyperplane that maximize the margin from nearest data vectors in both classes





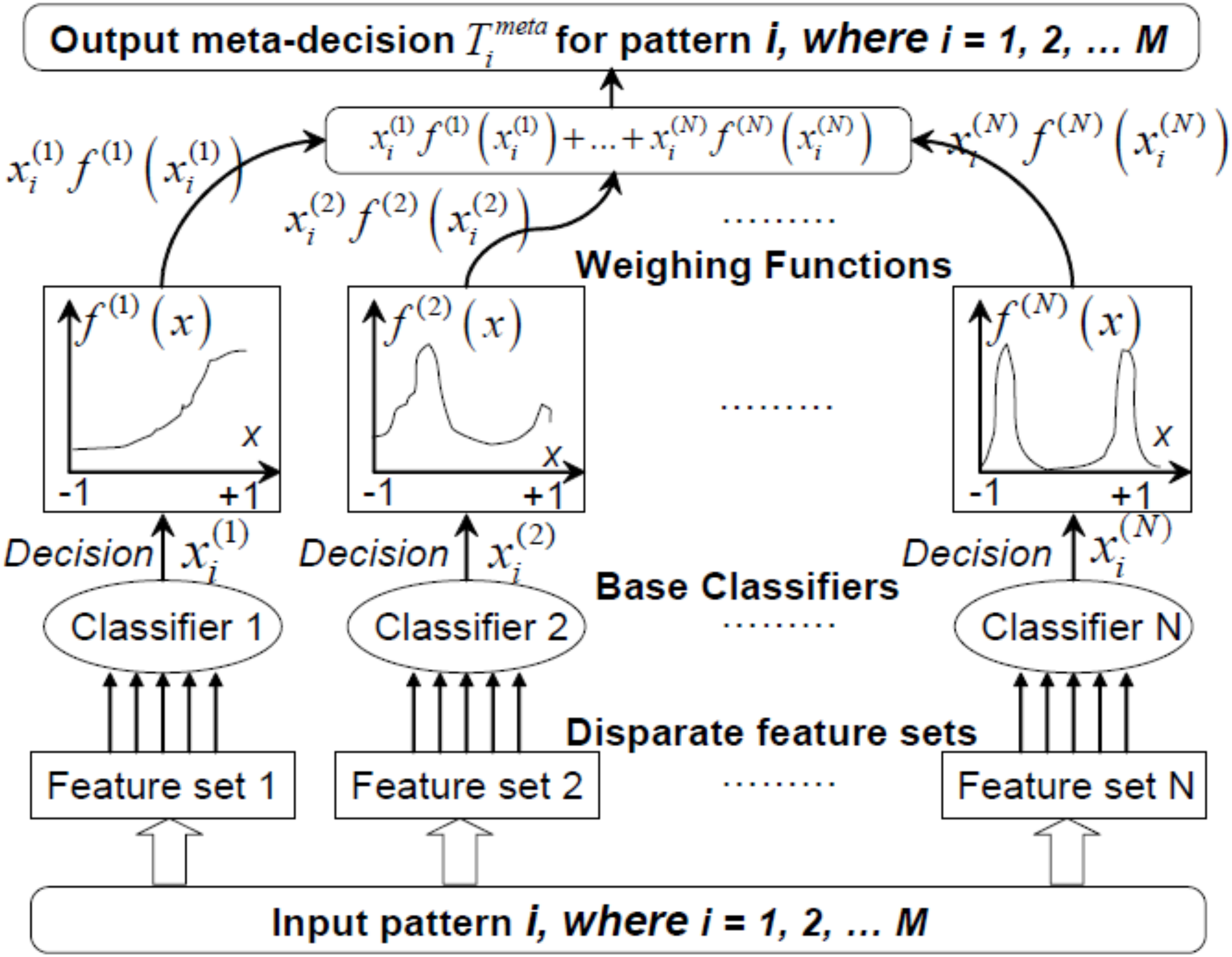
Unified Meta-Classification

- Combine different techniques with different strengths
- [Ding+ ASPDAC12]



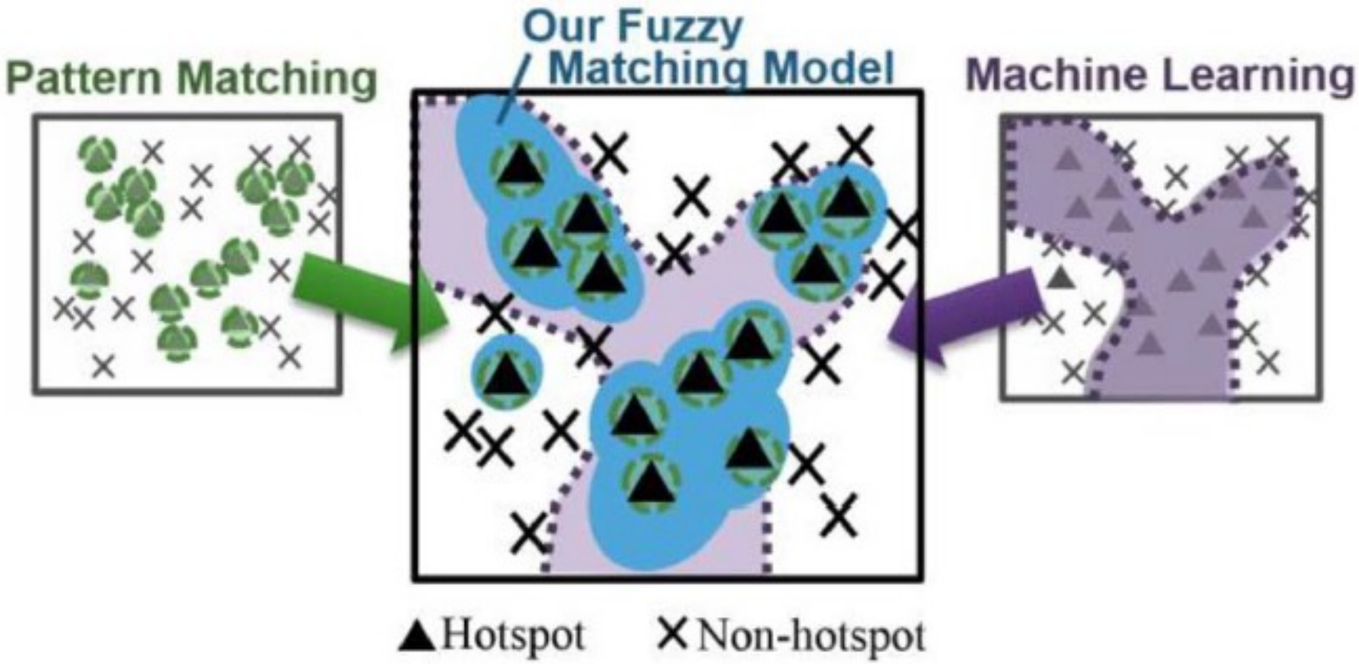
Unified Meta-Classification

- Final prediction based on weighted sum of base classifiers

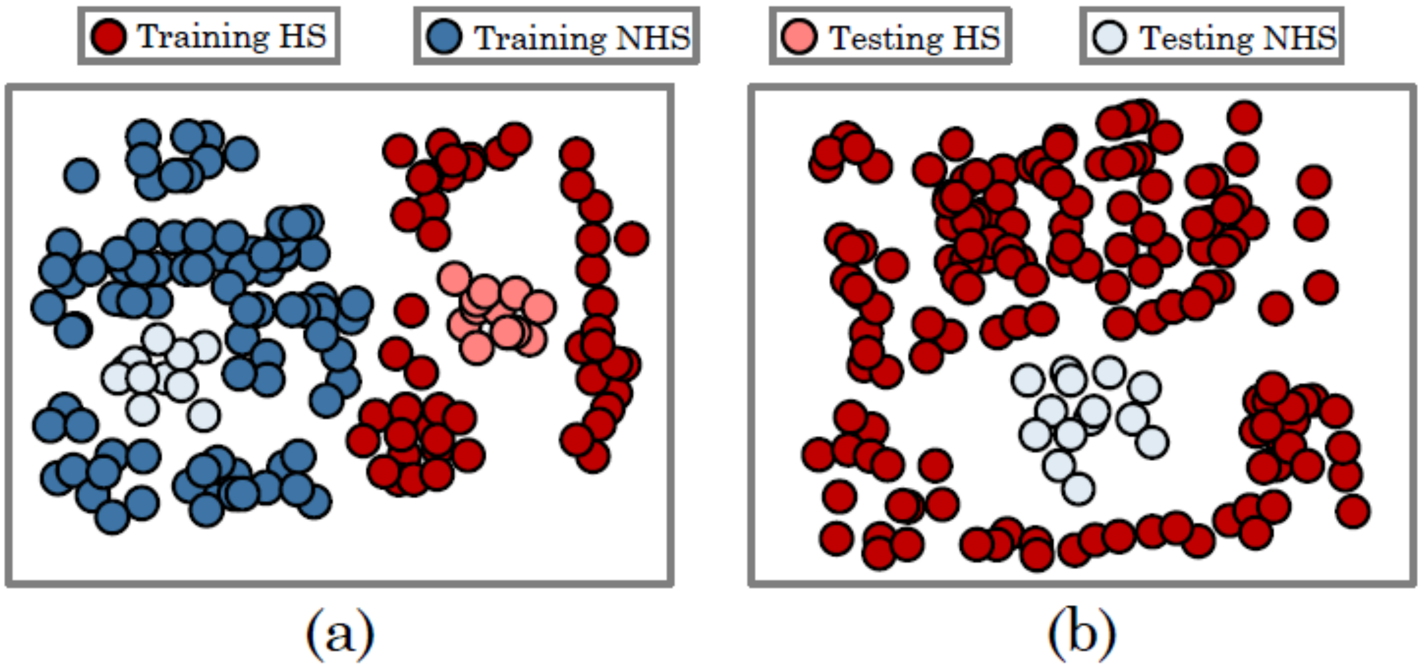


Fuzzy Matching

- Tune the fuzzy regions around known hotspots heuristically [Wen+ TCAD14]
  - Fuzzy regions will not contain non-hotspots
  - Locations close to high density non-hotspots are usually non-hotspots



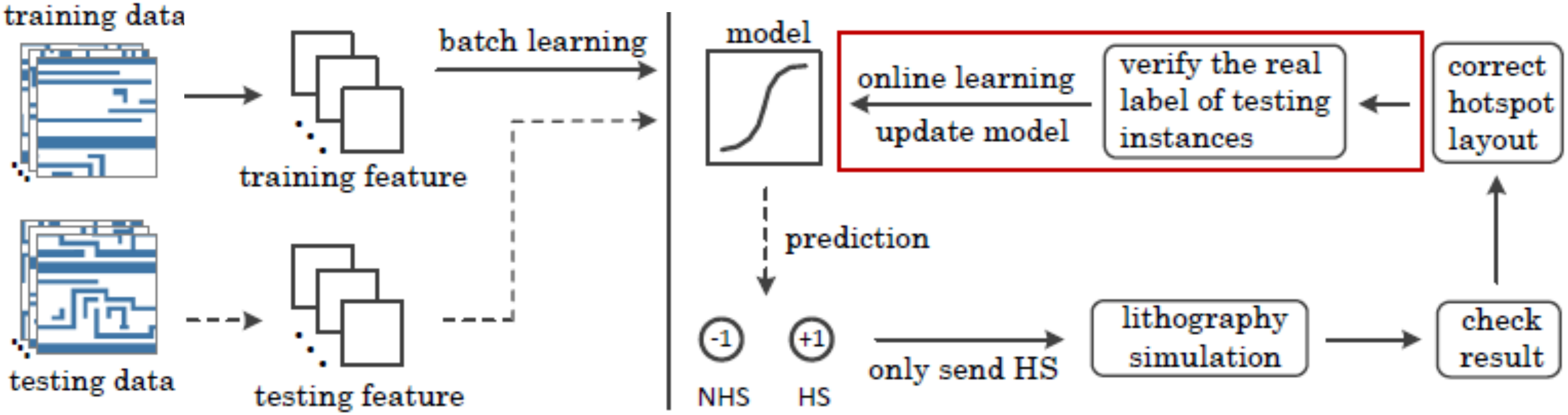
Online Learning



- We always want a situation like (a) but there is no guarantee that a situation like (b) will not occur
- With conventional machine learning method, the testing non-hotspots in (b) are likely to be wrongly predicted as hotspots
- To remedy this, we can dynamically update our model with instances predicted as hotspots whose labels are verified by simulation

Online Learning

- [Zhang+ ICCAD16]



- Instead of maximizing accuracy and reducing false alarms, we maximize accuracy and reduce overall detection and simulation time (ODST)
- Online learning is helpful to improve both accuracy and ODST

Deep Learning

- Extracting proper layout feature is hard
- Deep learning does not require manually crafted features, instead features are automatically extracted through neural network training
- CNN which has proved to be highly successful in pattern recognition and image classification tasks can be adapted for lithography hotspot detection [Yang+ SOCC17]

References

- [Ding+ ASPDAC12] EPIC: Efficient Prediction of IC Manufacturing Hotspots with a Unified Meta-Classification Formulation
- [Wen+ TCAD14] A Fuzzy-Matching Model with Grid Reduction for Lithography Hotspot Detection
- [Yu+ DAC12] Accurate Process-Hotspot Detection Using Critical Design Rule Extraction
- [Yu+ DAC13] Machine-Learning Based Hotspot Detection Using Topological Classification and Critical Feature Extraction
- [Zhang+ ICCAD16] Enabling Online Learning in Lithography Hotspot Detection with Information-Theoretic Feature Optimization
- [Zhang+ ISPD17] Bilinear Lithography Hotspot Detection
- [Yang+ SOCC17] Lithography hotspot detection: From shallow to deep learning