Portland State University

Approximate Pattern Matching using Hierarchical Graph Construction and Sparse Distributed Representation

Aakanksha Mathuria
(aakan@pdx.edu)

Dan W. Hammerstrom
(dwh@pdx.edu)

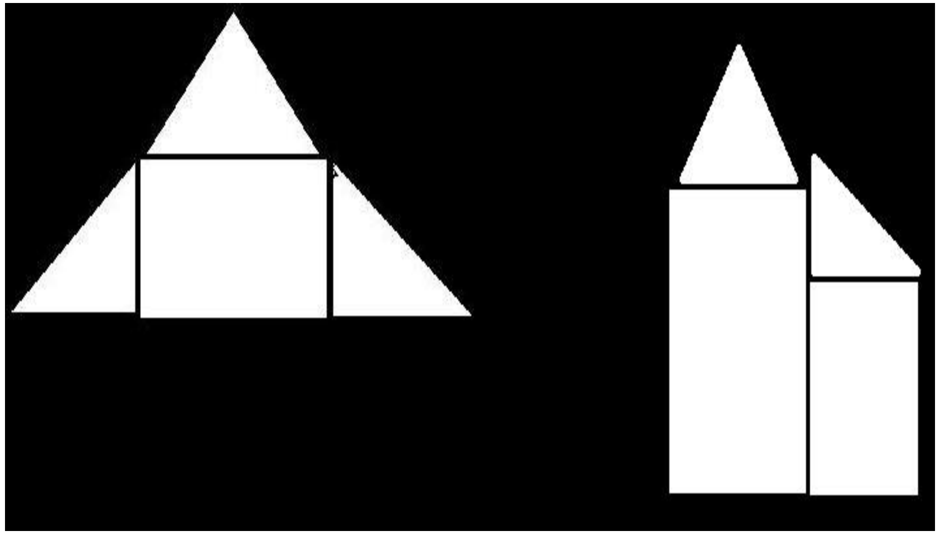
Problem Statement

- Real world image recognition involves understanding complex relationships among different features and/or among different components and subcomponents of objects.
- CNNs tend to lose relative position information during multiple levels of spatial pooling designed to provide position invariance.
    - This was one motivation for Geoff Hinton's "Capsule" networks
- Our approach use simplistic, cortical like networks using 2D arrays of associative modules based on Sparse Distributed Representations (SDR)
- Such arrays can be configured into hierarchies, which consist of:
    - Sparse distributed data representations => sparse activation => sparse connectivity.

Problem Statement (2)

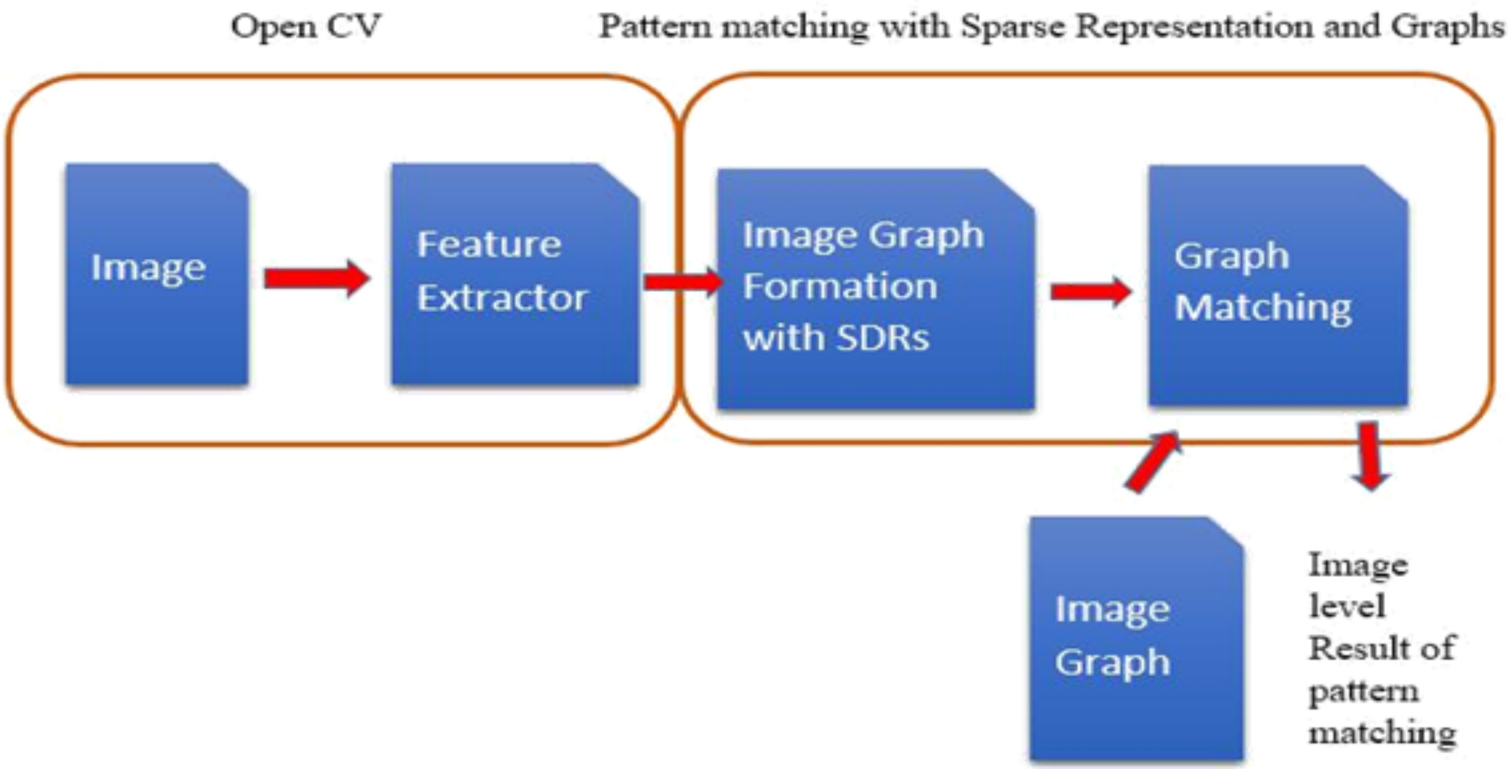
- We're Interested in the "back end", so we are keeping the "front end" processing minimal.
- For our initial experiments, we are using 2.5D (flat, but with possible occlusion) static "Blocks World" images.
- In the research described here, we are proposing a heuristic for solving approximate graph isomorphism to reduce the complexity of pattern matching by combining graph analytics and sparse distributed representations.
- This representation can be easily mapped to associative memory networks

- Here, we are taking simple block images such as triangle, rectangle etc. which can be combined in different ways to form complex images.

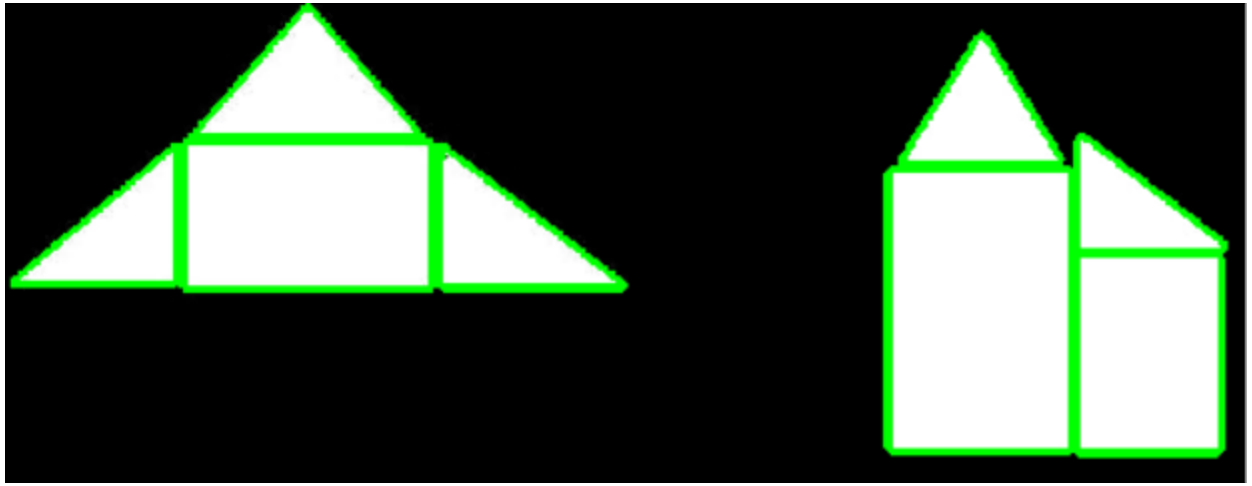


Simple Blocks-World image

Data Flow Pipeline



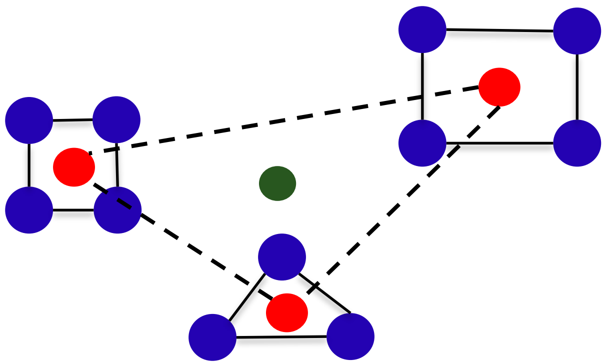
Object detection and feature extractor

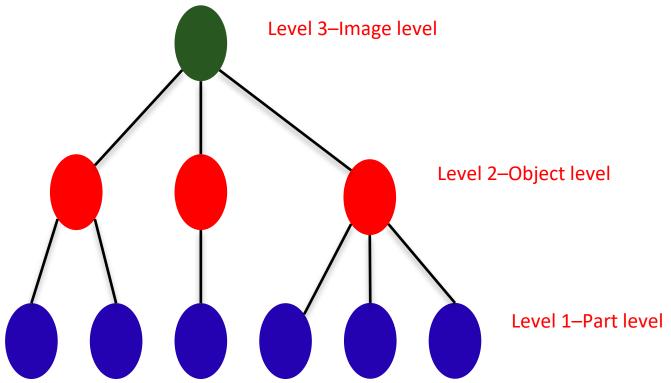


- Using OpenCV, we detect contours of the objects and features of the contours.
- Compute the object attributes from contour features such as centers, height, width, angle with the x and y-axis, etc.

Hierarchical Graph Construction

- Leverages Sparse Distributed Representation, which is inspired by cortical circuits.
- Efficient graph representations capture the structure of the objects and provide algorithmic benefits when recognizing complex images.
- A hierarchical graph for an image is constructed using a fixed-radius nearest neighbors algorithm.
- Compute objects by considering connected components from the graph.
- Currently using only three levels of hierarchy, though higher levels of hierarchy are possible.





Hierarchical Sparse Distributed Representation

- SDRs are determined bottom-up in the hierarchy.
- A single node's SDR stores its own information as well as its neighbors' (currently we are using one-hop connectivity).

SDR of a node: (length – 3991 bits)

1. Node's attributes
   1. Number of edges (10 bits)
   2. Height – width ratio (180 bits)
   3. Orientation angle (90 bits)
   4. Connectivity (11 bits)

   Sub-fields

2. Neighbors' attributes
   1. Number of edges (10 bits)
   2. Height – width ratio (180 bits)
   3. Orientation angle (90 bits)
   4. Relative position (angle with the node – 90 bits)

   Sub-fields

- Each bit in an SDR has semantic meaning.
- SDRs should be sparse (roughly the number of 1's should be the $\log_2$ of the dimension.).
- The use of SDRs should be mostly independent of the indexing scheme representing the graph.

Node's attributes				Neighbor 1's attributes				Neighbor 2's attributes			
Number of edges	Height-width ratio	Orientation angle	Connectivity	Number of edges	Height-width ratio	Orientation angle	Relative position	Number of edges	Height-width ratio	Orientation angle	Relative position
001…00	00…1…00	00…1…00	0010000	0001…00	00…1…00	00…1…00	00…1…00	00001…00	00…1…00	00…1…00	00…1…00

Length of SDR: $l = s + b + c + c\,(s + 2b)$

Number of 'ON' bits: $o = 1 + w + 1 + c\,(1 + 2w) = (2 + w) + c\,(1 + 2w)$

s = Number of edges
b = Height – width ratio
c = Connectivity
w = Number of active bits

- Higher levels are determined by taking 'union' of the lower level's connected nodes.
- By the union property, a single SDR is able to store a dynamic set of elements.

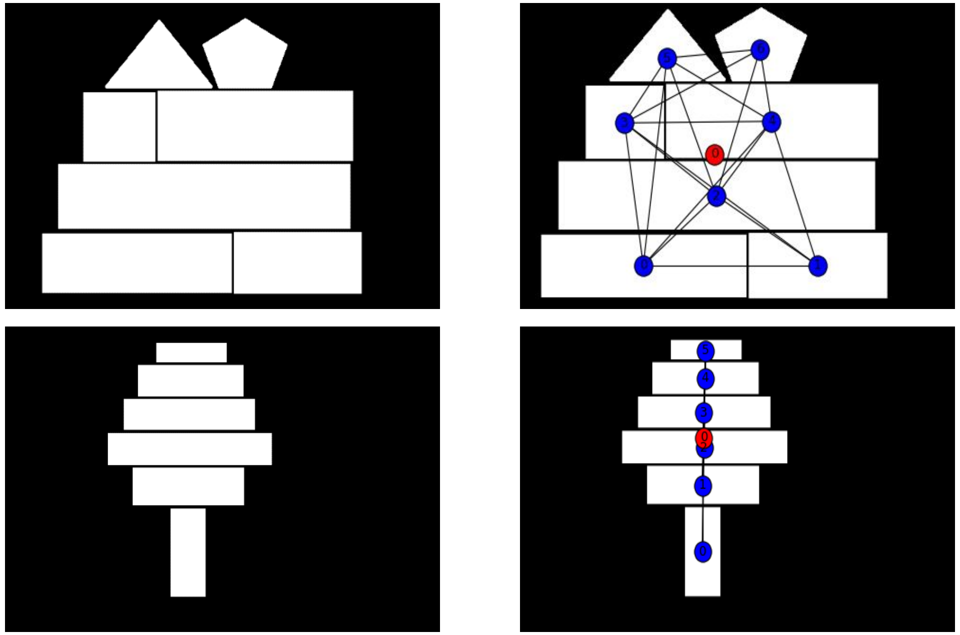
Graph Matching

- The graph matching is done by taking dot product of the graph SDRs, which is the basic computation of a large Associative Memory model:

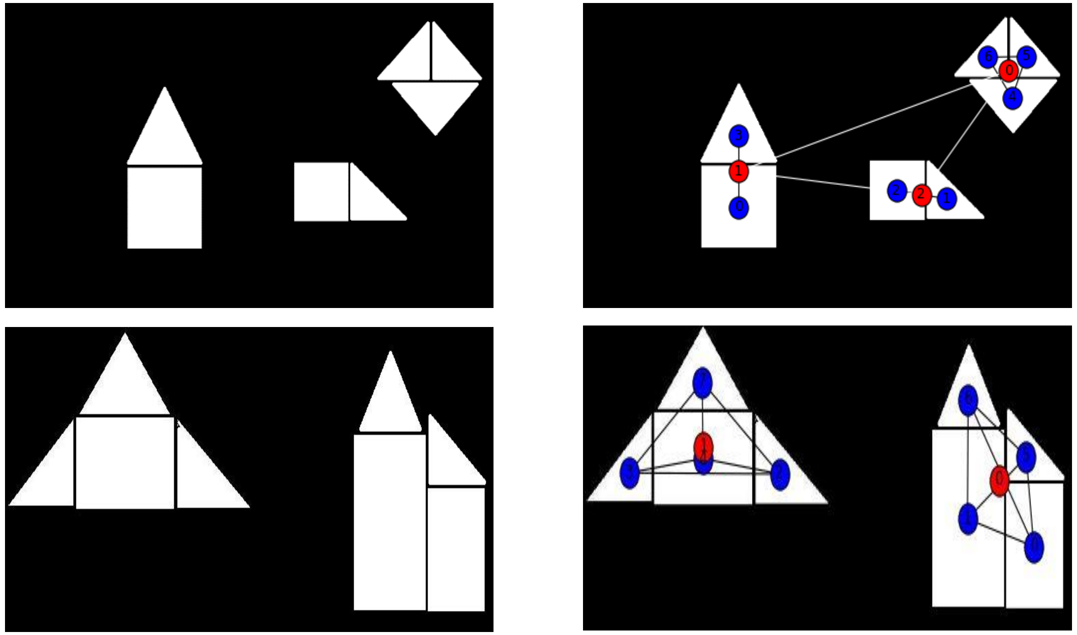
$$match\ (S_A, S_B) \equiv overlap\ (S_A, S_B) \equiv S_A\ .\ S_B \geq \Theta$$

- We demonstrate the approximate graph matching in *O(1)* and by choosing k nodes' subgraph out of n nodes' big graph in $O(n^k)$, subgraph matching in *O(1)* instead of solving in non-polynomial times with the help of SDR properties.
- Image graphs tend not to be too large or complex, as opposed to knowledge representation, for example.
- The sub-graphs respect the hierarchy. Graph processing will involve small, local neighborhood graphs and not an entire image.

Results

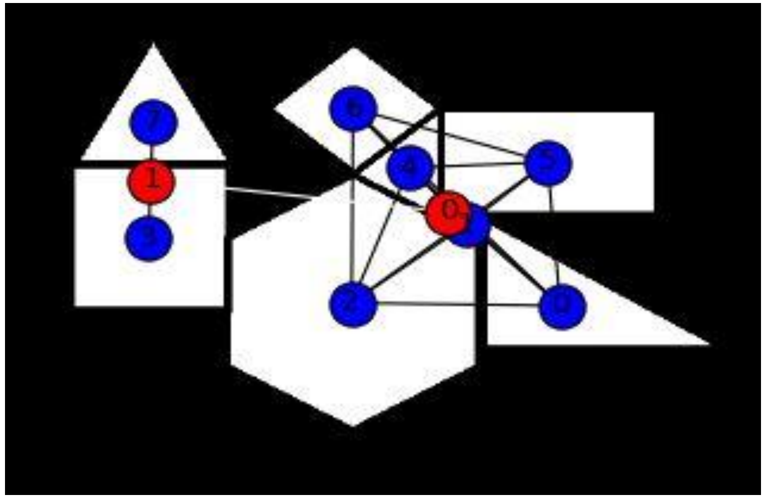


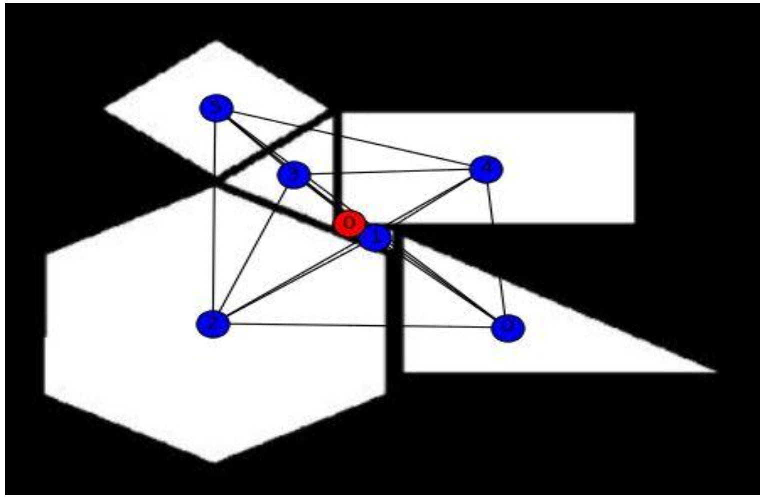
2D Blocks World images with one object and generated graphs.



Images with multiple objects and their generated graphs

Two graphs with sub-graph isomorphism





- The SDR overlap exceeds the threshold for level 2.

Conclusion

- Our approach captures the connectivity information in images
  - Hierarchical object detection.
- Hierarchical graph construction of objects with the Euclidian distance criteria.
- Approximate Graph isomorphism (graph matching) and sub-graph isomorphism with this technique results in $O(1)$ and $O(n^k)$ complexity.
  - Compared to be solvable in non-polynomial time.

Future Work

- Increase size of random images to more accurately assess False Positive rates with realistic graphs
- Develop cortical like models that approximate the computation of the SDRs proposed here

Thank you!