UCRL-PRES-226096

ENGINEERING

Turning Concepts into Reality

Algorithms for Fast, Robust Model-Based Polygon Detection

Siddharth Manay

CASIS 2006, 16-17 Nov.

UCRL-PRES-226096

Work performed under the auspices of the U.S. Department of Energy by the University of California,
Lawrence Livermore National Laboratory under Contract No. W-7405-Eng-48.

Unclassified

Polygons in Images

- Need to search for objects in large images.
- Why Polygons?
  - Polygons are useful general purpose descriptions of many different types of objects in images.
- Key design choices:
  - In lit, polygons are often constructed from detected *edges*. We construct polygons from detected *corners.*
  - We use *models* to allow user to constrain search.

Image courtesy of CA Spatial Information Library
http://archive.casil.ucdavis.edu/casil/

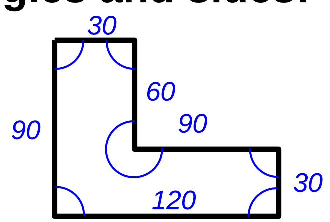
Polygon Models Used to Constrain Search

- **3 Polygon models provide constraint/flexibility tradeoff.**
  - **Choose the one that's right for the application.**

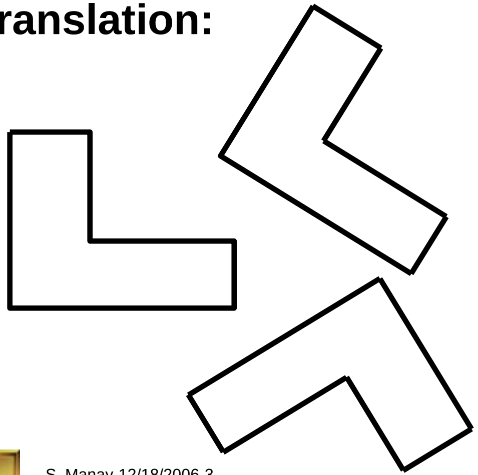
Constrained ⟵⟶ Flexible

Absolute Side-Length Model

•Angles and sides:

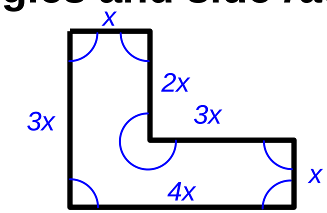


•Allows rotation and translation:

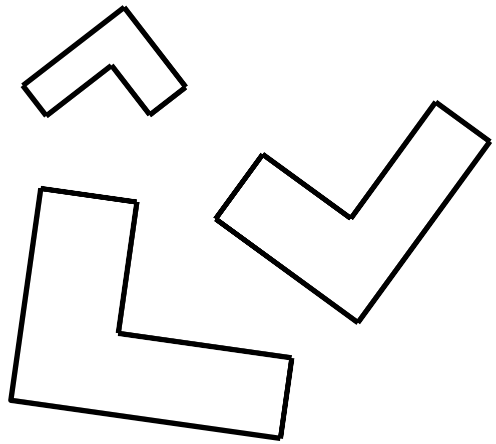


Relative Side-Length Model

•Angles and side *ratios:*

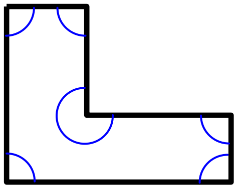


•Allows rotation, translation, and scaling:

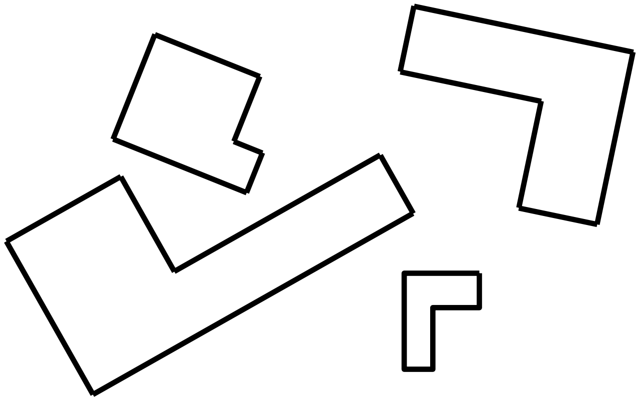


No Side Length Model

•Angles *only:*



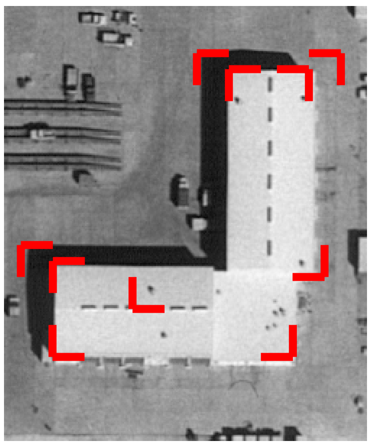
•Allows rotation, translation, scaling, and other distortions:



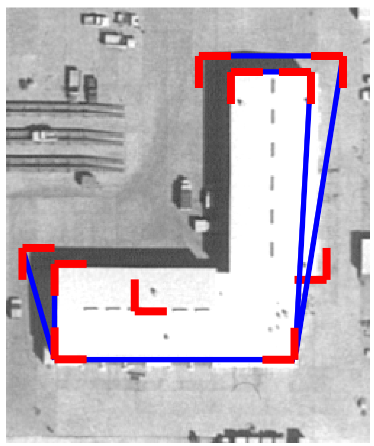
Extracing Polygons: Overview

- **Processing Pipeline:**

Detect corners

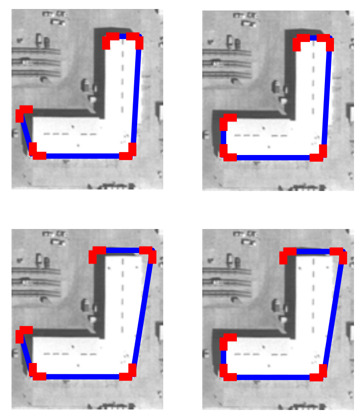


Grow corner trees

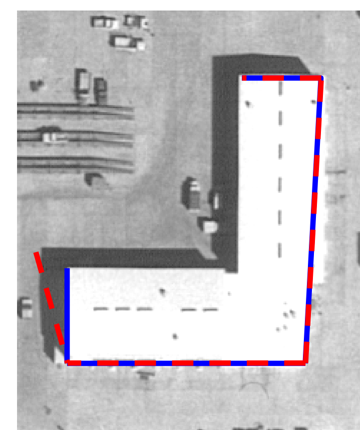


Extract Complete and Incomplete polygons from trees

Evaluate Polygons:
- **Model fit**
- **Image fit**
- **Other filters**

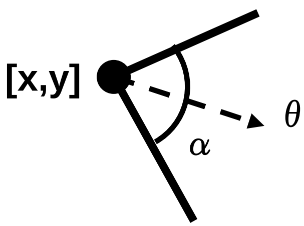


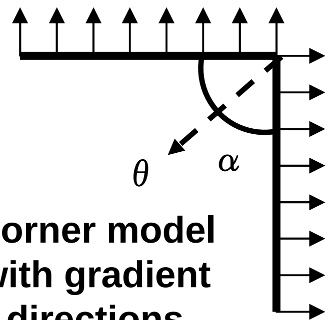
Disambiguate collisions.



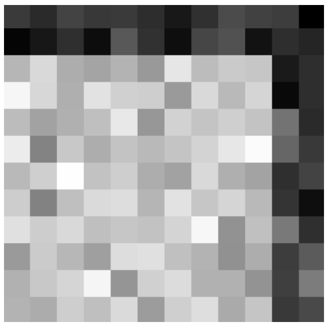
Detect corners with GDM

- Corners are modeled as location, acuteness, and orientation.



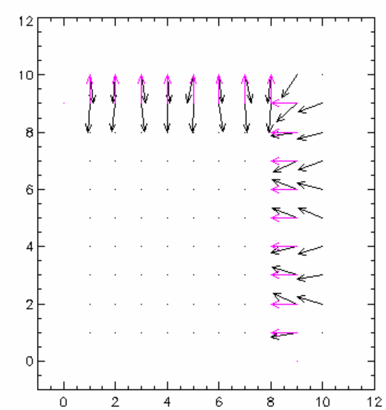


Corner model with gradient directions

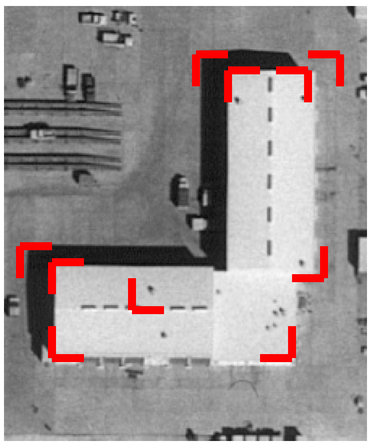


Corner in image

- Gradient Direction Matching is used to extract corners in images:



Search and Evaluate match between image and model gradients

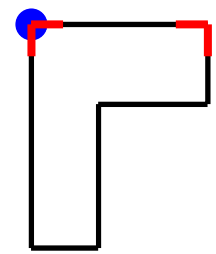


Detect corners

Corner Search

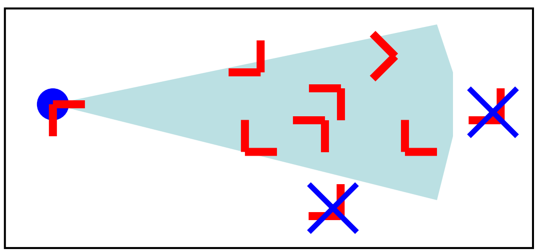
- Given a corner, a model, and the correspondence between the two, search for next corner is well constrained.
  - Corner orientation/acuteness model allows fan or circle constraints:

Model:

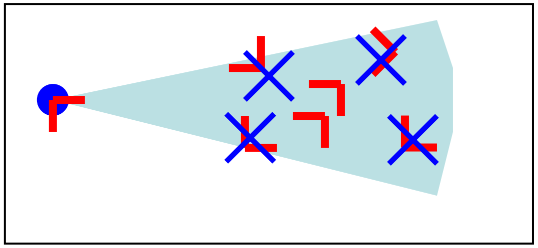


Case I: no side length information.

1) Search fan region:

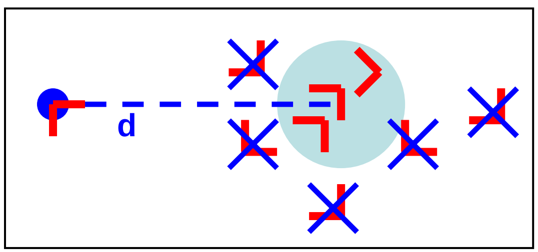


2) Search by orientation, acuteness:

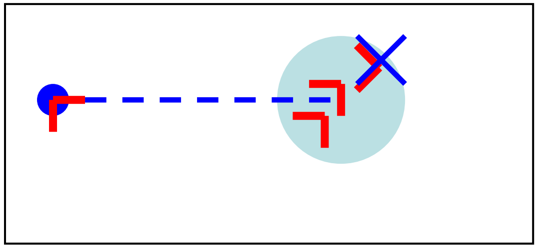


Case II: side length information.

1) Search circle region:



2) Search by orientation, acuteness:



Grow corner trees

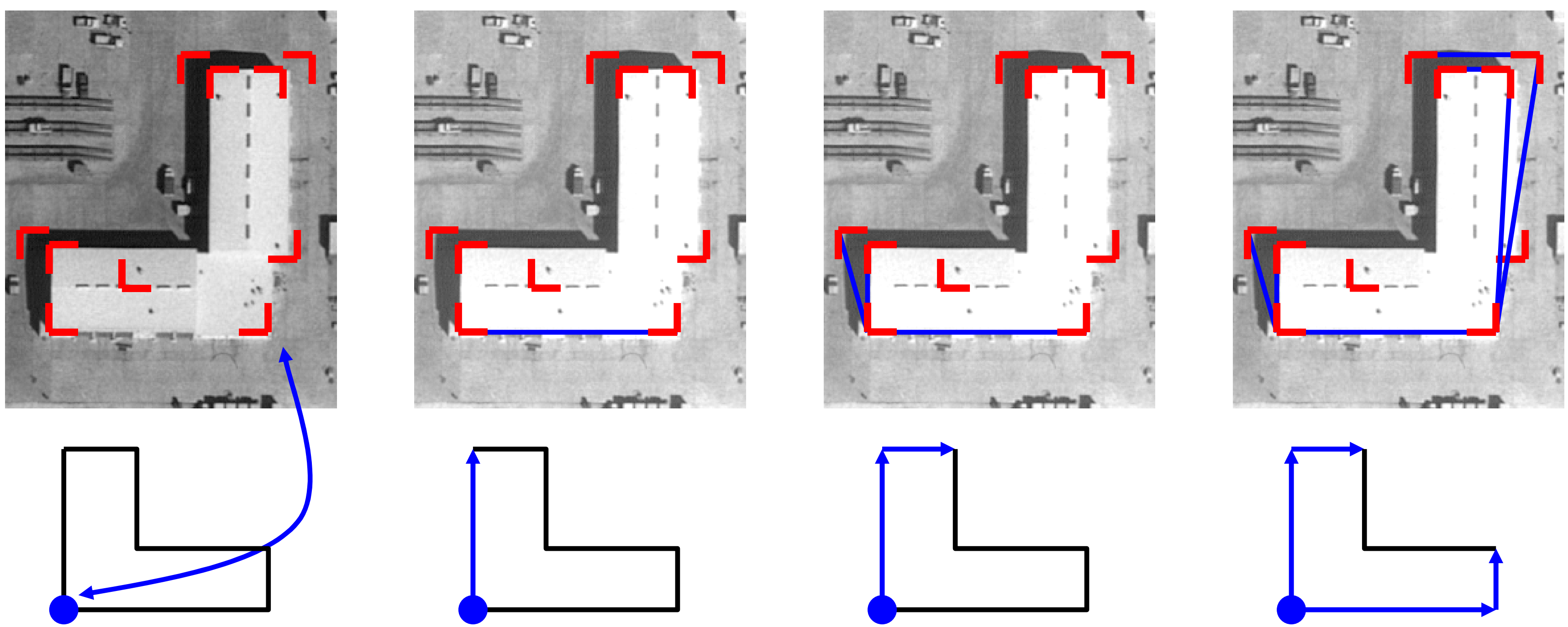
- **Using polygon model constraints, connect corners into trees.**

Start with one corner and model correspondence…

Using model, find next corner…

Continue CW until there are no more corners.

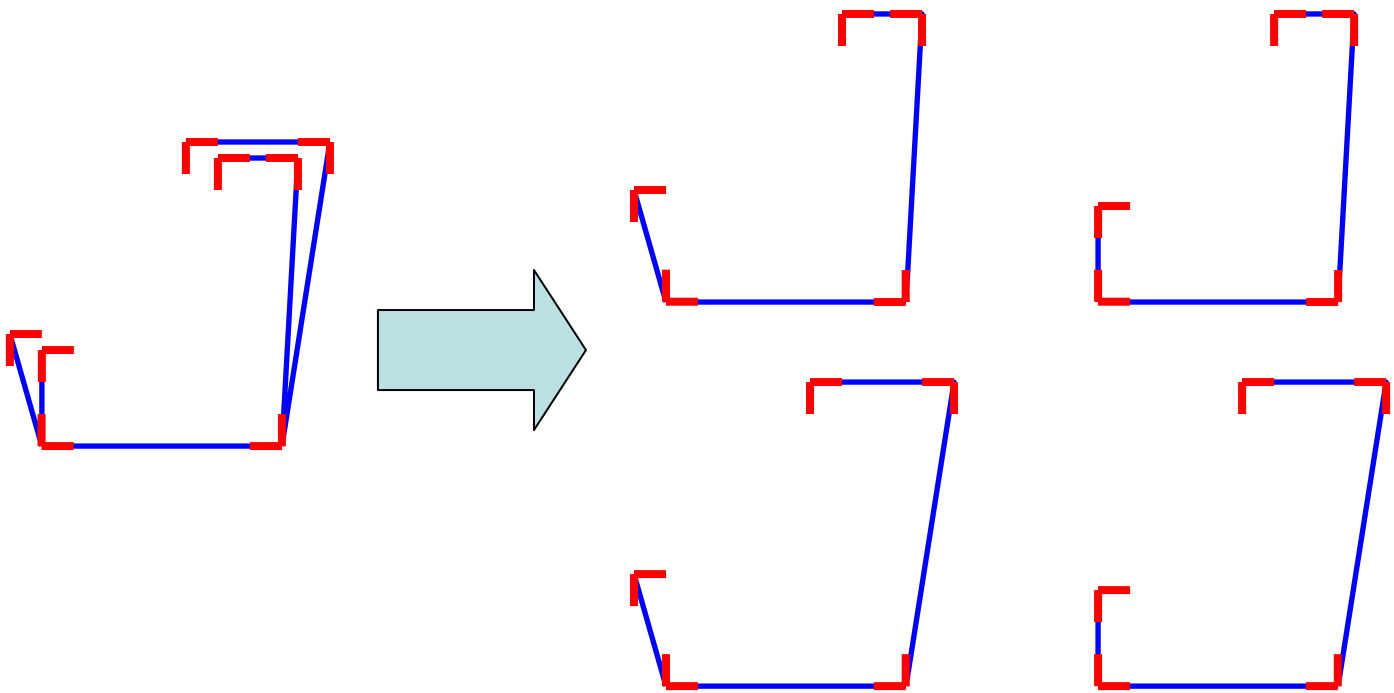
Trace CCW as well.



Model + first corner gives orientation, acuteness, and (maybe) position information for next vertex.

Extract Polygons

- Convert trees to Complete and Partial Polygons.
  - Complete: polygon is closed.
  - Partial: corners are missing.
- Paths in corner trees are possible (partial) polygons:



Unclassified

Evaluate Polygons

- **Apply size, geometry filters.**
  - **Min/max area, min/max perimeter, min/max side length, etc…**
- **Score based on geometric fit; discard if score is less than a threshold.**
  - **Score based on corner angles:**

$$E_{angle} = \frac{1}{n\pi} \sum_k |\alpha^*_k - \alpha_k|$$

  - **Score based on side lengths (if applicable):**

$$E_{rel\ side} = \frac{1}{n} \sum_k \left| \frac{l^*_0 l_k}{l_0 l_k^*} - 1 \right| \qquad E_{abs\ side} = \frac{1}{n} \sum_k \frac{|l^*_k - l_k|}{l^*_k}$$

- **Score based on image data; rank based on score.**
  - **Gradient direction score:**

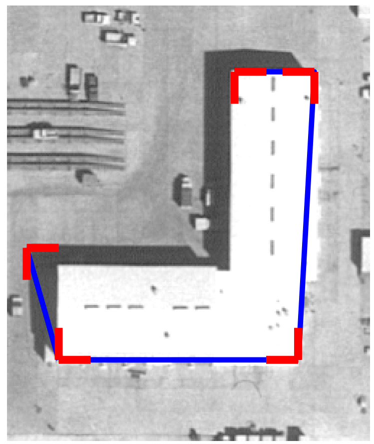
$$S = \frac{\sum_{x,y} w(x,y) a(x,y) \cos^2(\angle \nabla I(x,y) - \beta)}{\sum_{x,y} w(x,y)}$$

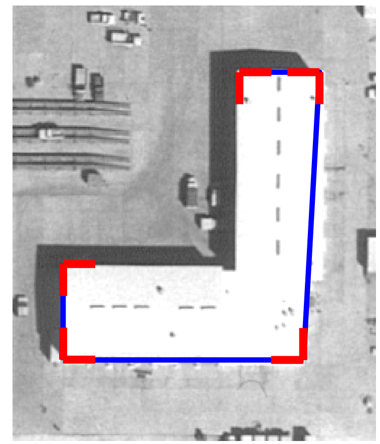
$$a(x,y) = \begin{cases} 1 & |\nabla I| > C_{min} \\ 0 & \text{else} \end{cases}$$

  - **Other measures could include region homogeneity, texture,…?**

Evaluate Polygons

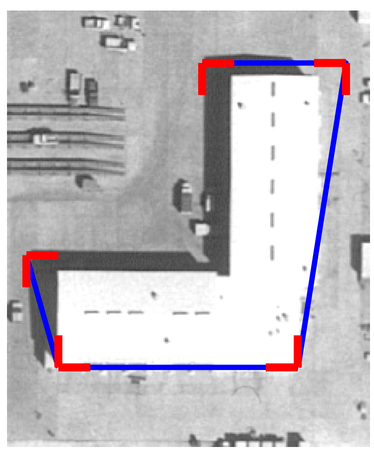
Weak geometry, Okay gradient

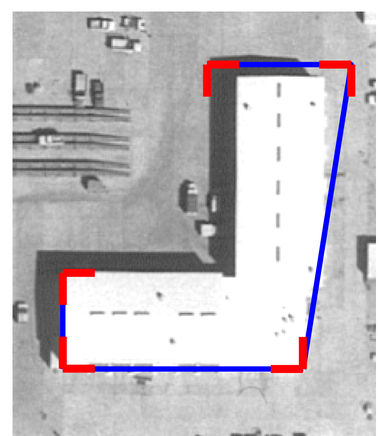




Okay geometry, Okay gradient

Bad geometry, Bad gradient



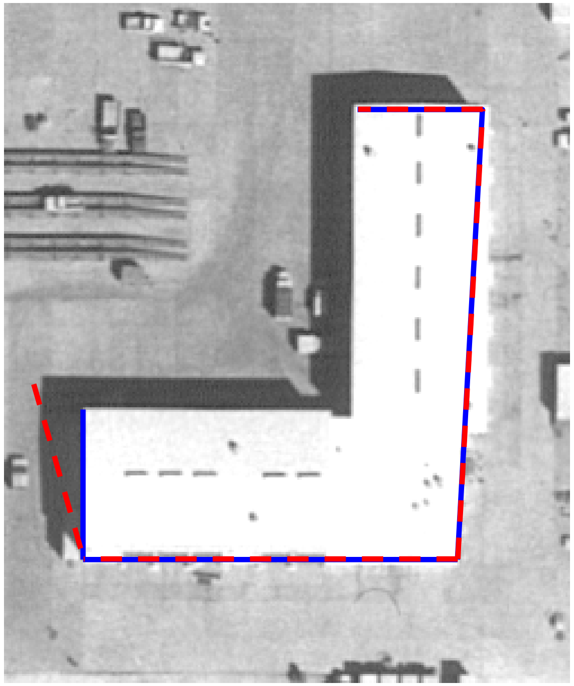


Weak geometry, Bad gradient

Unclassified

Disambiguate

- If polygons overlap, conflicts are resolved.
  - Which polygon is more complete?
  - Which has the stronger gradient direction score?

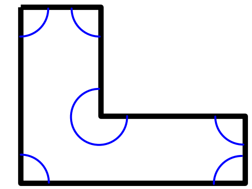


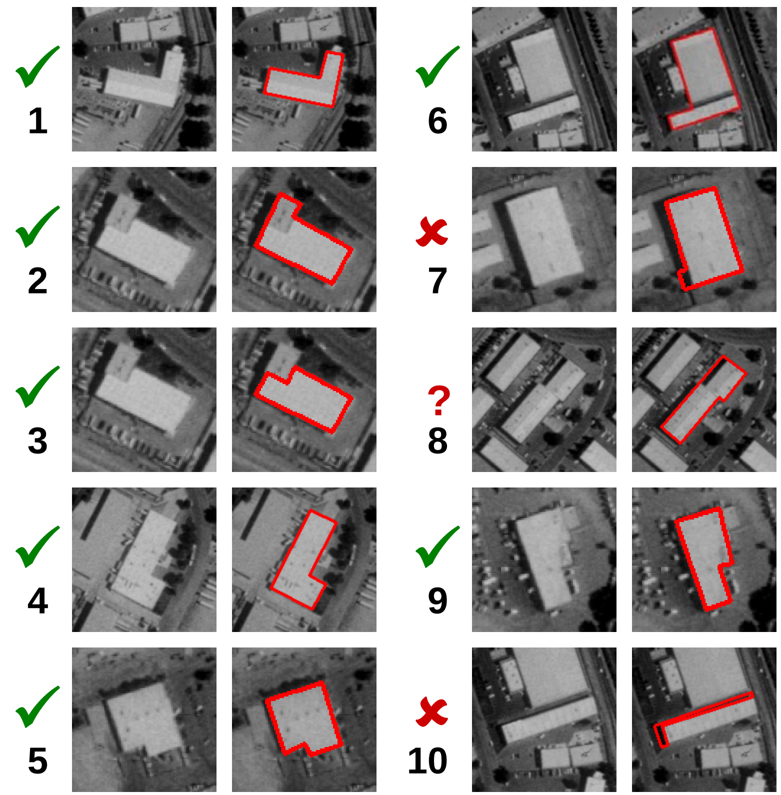
- If polygon is partial, we can still use it to identify the object.
- Reconstruction is also possible using model.

Unclassified

No-Side-Length Model Example

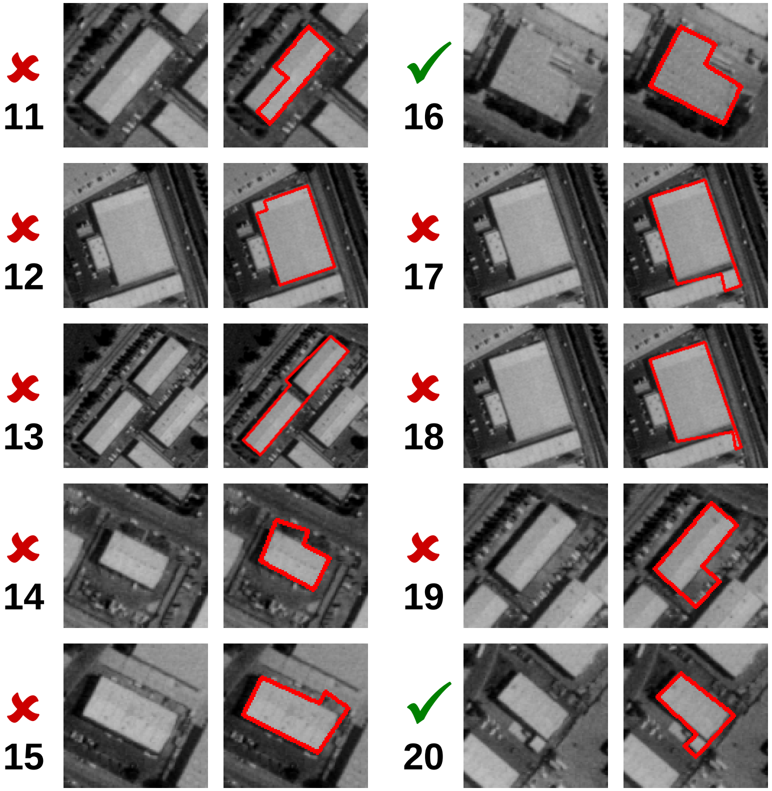
- "L"-shaped model with 6 corners *only*.
- Hits 1-10





No-Side-Length Model Example

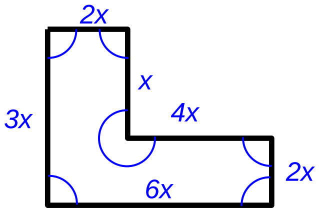
- **Hits 11-20**



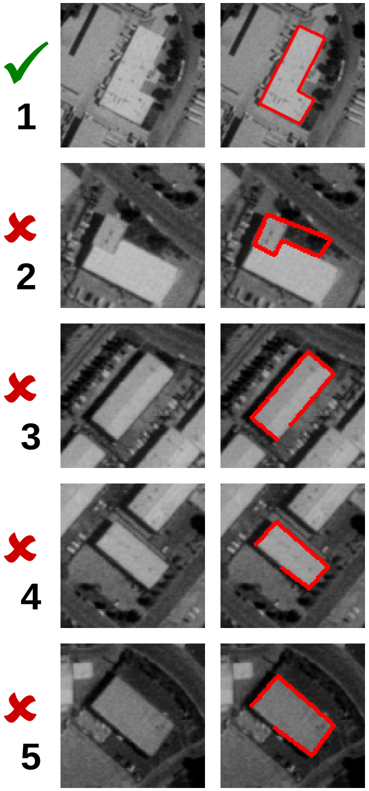
Unclassified

Relative-Side-Length Model

- “L”-shaped model with 6 corners and side *ratios*.

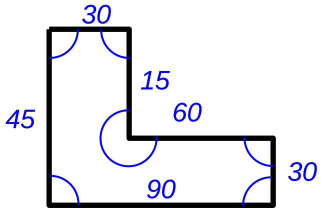


- Hits 1-5



Absolute-Side-Length Model

- "L"-shaped model with 6 corners and side lengths.



1

2

Polygon Model Matching

- Uses a family of flexible models.
  - Choose model best suited to application.
- Detects corners using Gradient Direction Matching.
- Assembles corners into polygons.
  - Assembly is constrained by the model.
  - Corners are organized into corner trees.
  - Robust to missing corners.
- Polygons are filtered based on match to model, other parameters.
- Polygons are scored using Gradient Direction for final ranking.