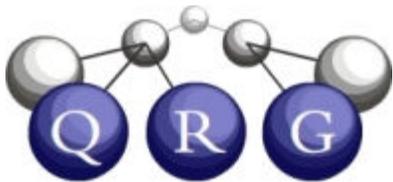
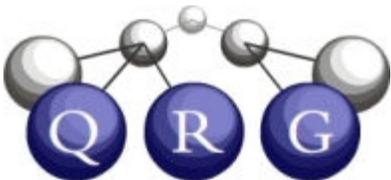


Visual Processing in CogSketch



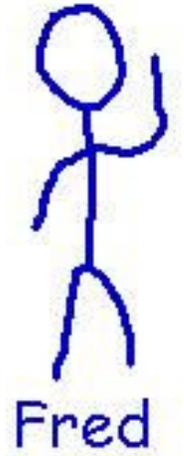
Some Preliminaries

- Visual versus Spatial relationships:
 - Visual relationships: Computed over glyphs.
 - Spatial relationships: Hold between what is denoted by the glyphs
 - Visual relationships + genre + pose \rightarrow Spatial relationships
- Our visual computations are inspired by psychological evidence when available
 - Best guesses otherwise
 - We expect it to continue to evolve

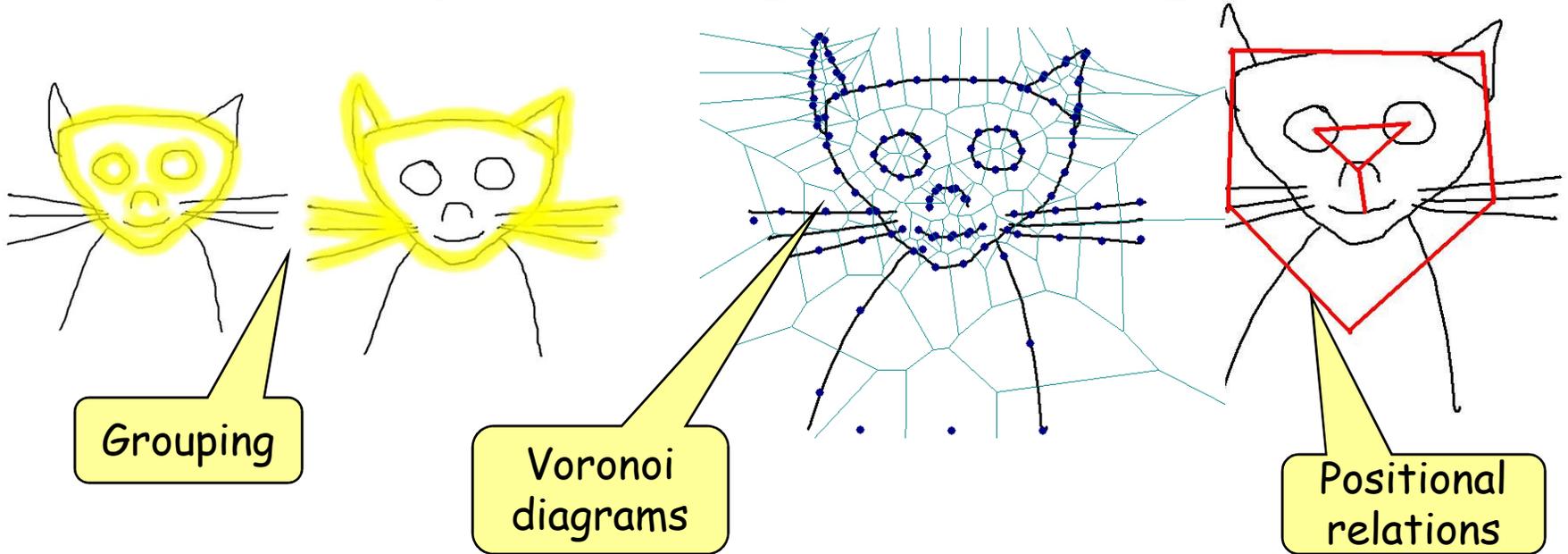


Glyphs

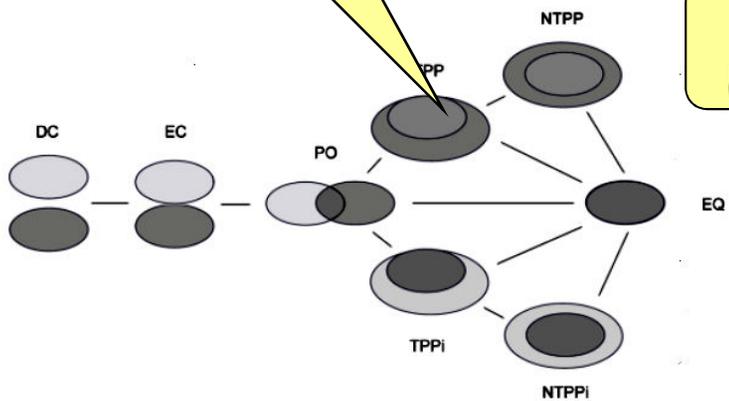
- Glyphs have two parts: *Ink* and *Content*
- Content = the entity represented by the glyph
 - Instance of some collection in the KB
- Ink = visual representation of the content
 - Consists of all of the ink drawn between button presses
- Visual properties are computed on the ink
 - Only coarse visual properties computed automatically
 - Bounding box
 - Closed contour (ink needn't be connected)
 - Major/minor axes
 - Small set of visual relationships between glyphs
 - Segmentation, other visual relationships computed on demand (e.g., perceptual sketchpad)



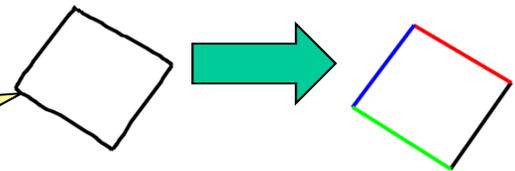
Some CogSketch spatial computations



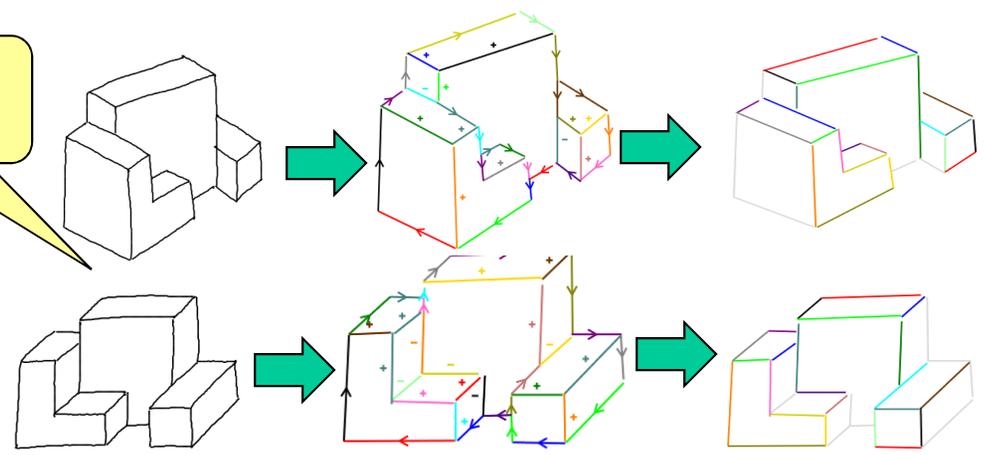
Qualitative Topology



Shape decomposition

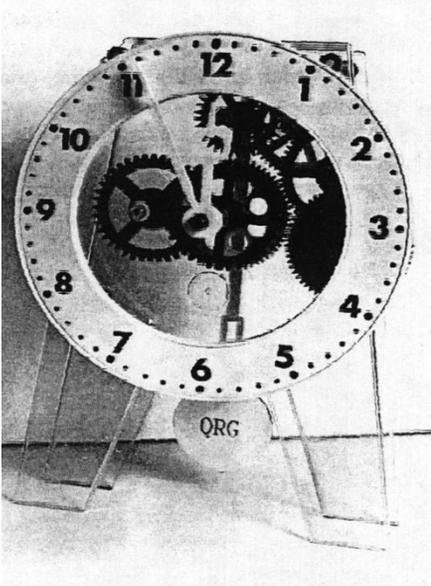


Mental Rotation

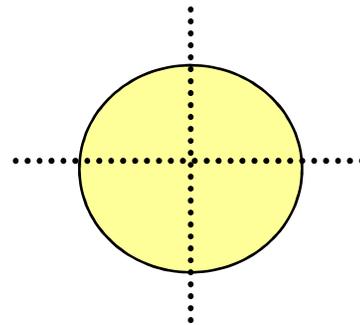


Qualitative Spatial Reasoning

- Claim: Symbolic vocabularies of shape and space are central to human visual thinking (cf. Forbus 1980; Forbus, Ferguson & Usher 2001)
 - They are computed by our visual system
 - Their organization reflects task-specific conceptual distinctions and conventional symbol systems as well as visual distinctions
 - They provide the bridge between conceptual and visual representations

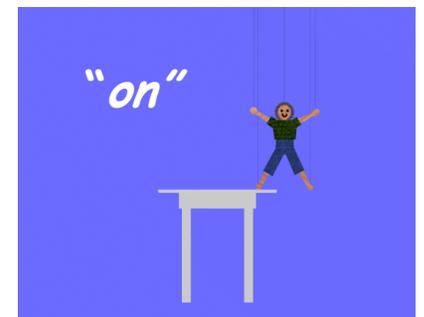


**CLOCK
project
(Forbus,
Nielsen, &
Faltings 1991)**



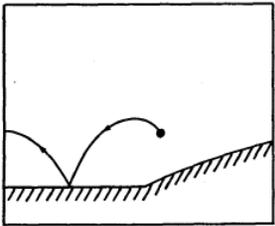
**Spatial categories affect
location judgments
(Huttenlocher & Hedges)**

**Spatial language
affects retrieval
(Gentner & Feist,
2001)**

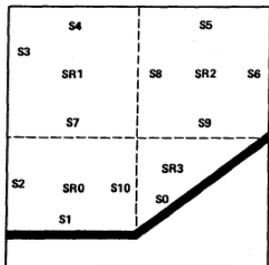


Metric Diagram/Place Vocabulary model

- Metric Diagram: Quantitative, visual representations and processing
- Place Vocabulary: Task-specific qualitative representations of shape and space, grounded in the metric diagram



FROB (Forbus, 1980)



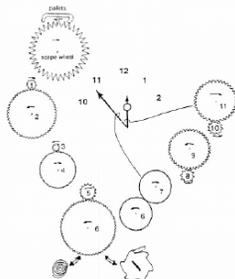
SREGION 0
 left: SEGMENT 2
 right: SEGMENT 10
 up: SEGMENT 7
 down: SEGMENT 1
 class: SREGION

 SEGMENT 1
 up: SREGION 0
 connecting-region: SREGION 0
 class: SURFACE

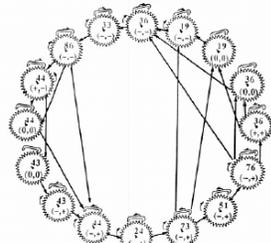
 SEGMENT 2
 right: SREGION 0
 left: SPATIUM-INCOGNITO
 connecting-region: SREGION 0
 class: BORDER

 SEGMENT 10
 left: SREGION 0
 right: SREGION 3
 class: FREE

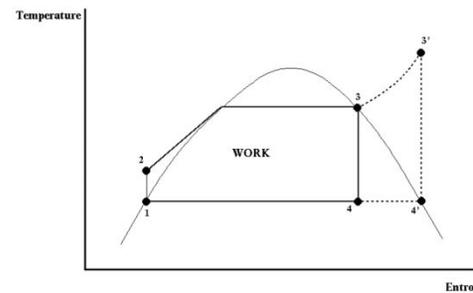
CLOCK Project: Example



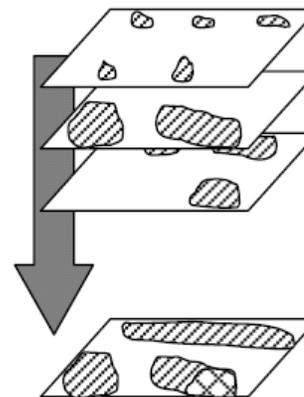
Input: Annotated diagram



Output: Set of possible behaviors, both desired and potential malfunctions



SKETCHY (Pisan, 1994)



Hydrography

Vegetation

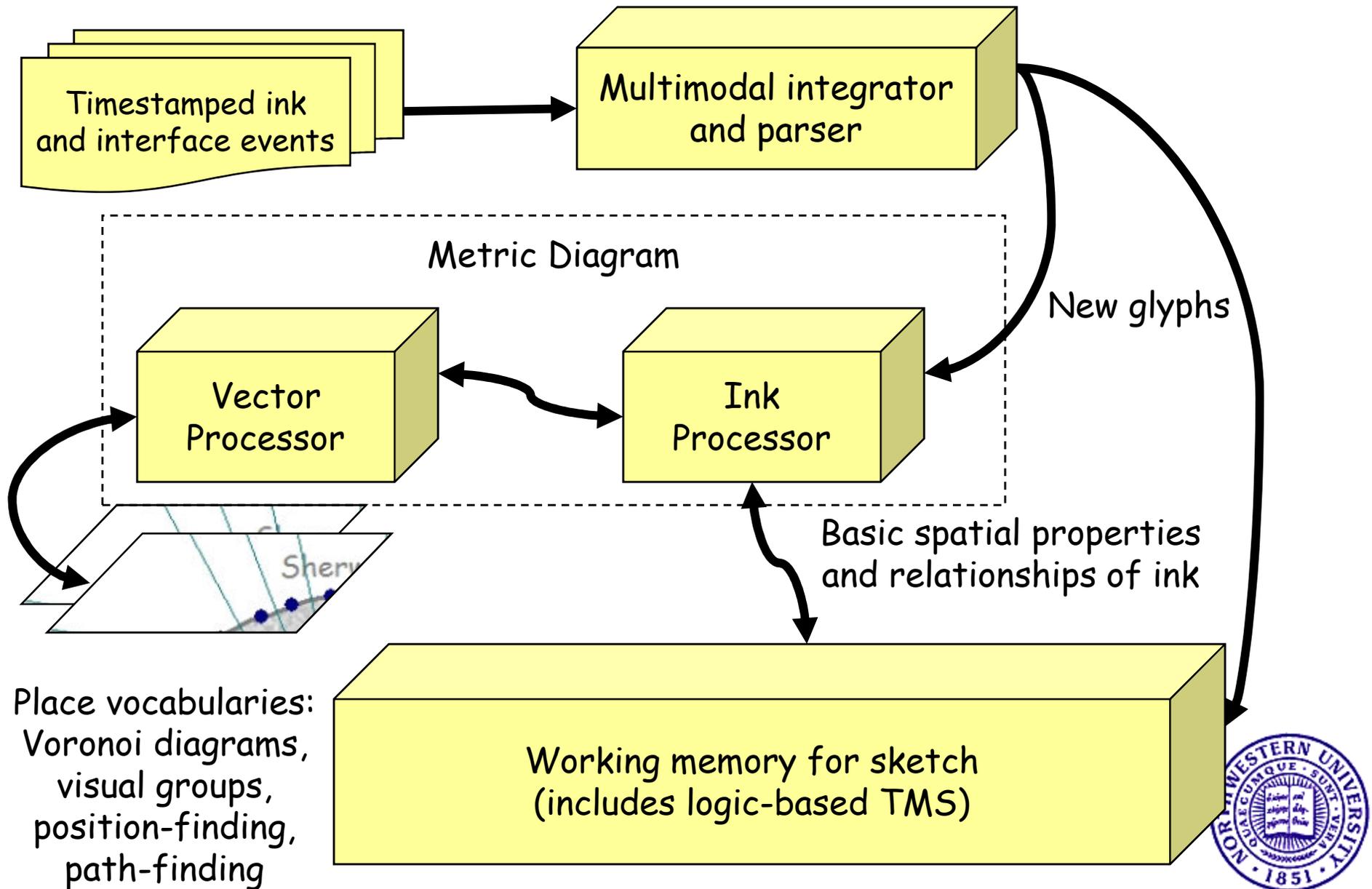
Slope

Etc...

Combined
Obstacle
Overlay

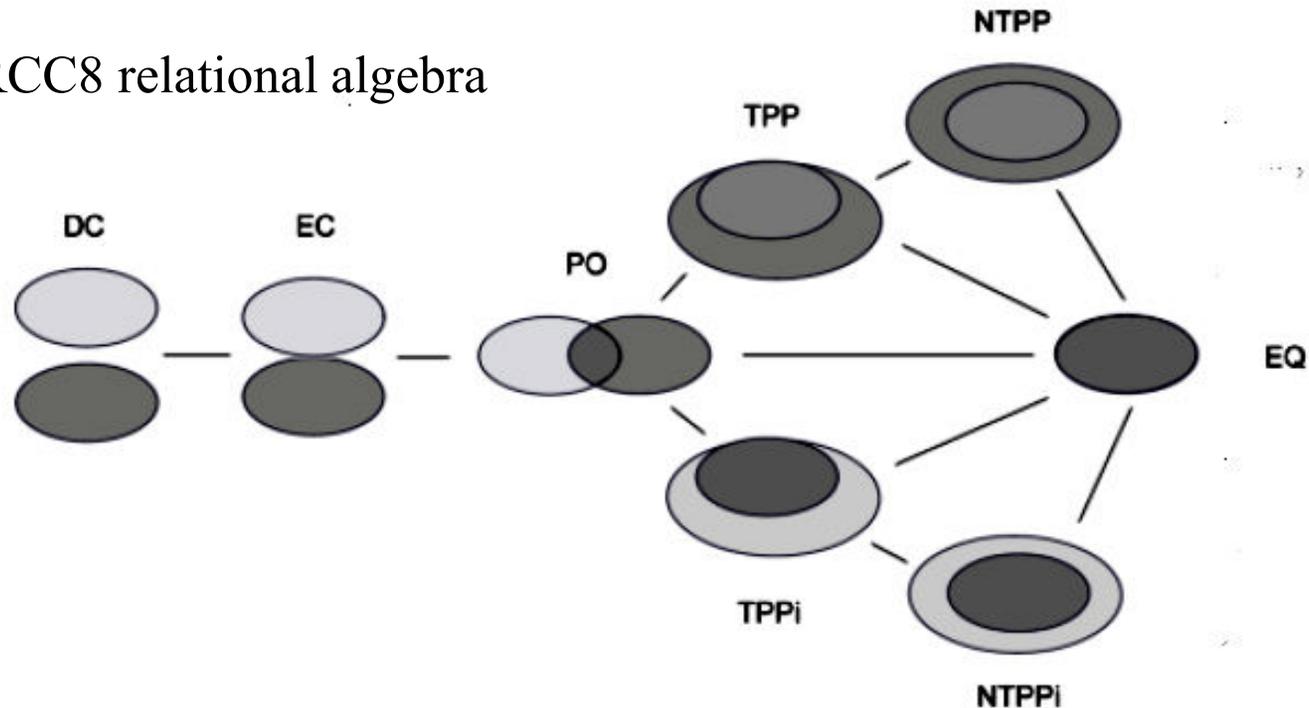
GIS-based
 Trafficability
 Reasoner
 (Donlon &
 Forbus, 1999)

Spatial Reasoning in CogSketch

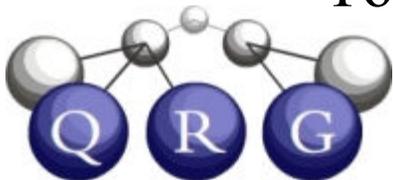


Qualitative Topology

Cohn et al's RCC8 relational algebra

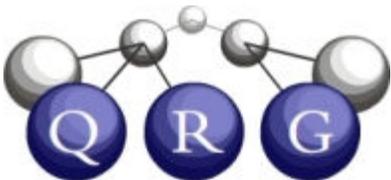


- Provides natural vocabulary for some visual concepts
 - Containment: NTPP, TPP
 - Touching: PO, EC



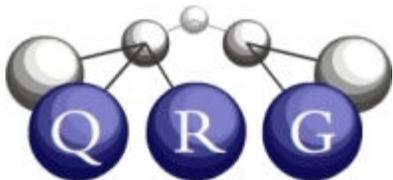
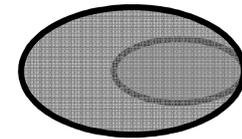
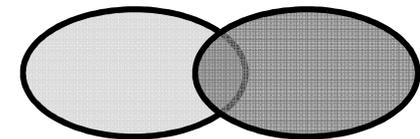
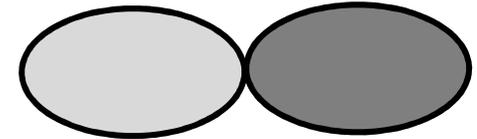
Using RCC8

- Compute relationships directly from ink
 - Transitivity algebra unnecessary
 - Need to be clever about noise
- Computed between every pair of glyphs on a layer
 - Incrementally updated when a glyph is moved or resized
 - Only computed across layers on demand
- Internal uses
 - Controlling computation of other relations
 - Positional relations aren't computed when there's containment
 - Direct inference of other topological relations



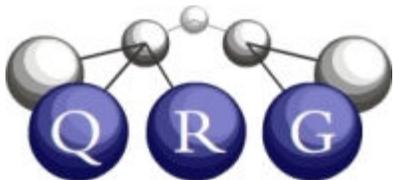
Higher-Level Topological Relations

- `objectsIntersect`
 - The ink of the two glyphs intersects
 - `objectsOverlap`
 - The interiors of the two glyphs overlap
 - `objectContains`
 - One glyph lies within another glyph's area
-
- Not mutually exclusive
 - Used in comparison



Contained Glyph Groups

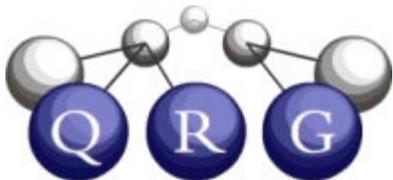
- When more than one glyph is NTTPi, TPPi of some other glyph
 - Single-level, groups can be found recursively
- (ContainedGlyphGroupFn
 (GlyphFn Object-9 User-Drawn-Sketch-Layer-1)
 (TheList (GlyphFn Object-15 User-Drawn-Sketch-Layer-1)
 (GlyphFn Object-16 User-Drawn-Sketch-Layer-1)
 (GlyphFn Object-19 User-Drawn-Sketch-Layer-1)
 (GlyphFn Object-20 User-Drawn-Sketch-Layer-1)))



Connected Glyph Groups

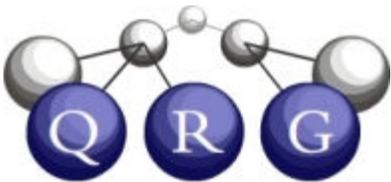
- Set of glyphs connected via EC or PO
- (ConnectedGlyphGroupFn
 (TheList (GlyphFn Object-10 User-Drawn-Sketch-Layer-1)
 (GlyphFn Object-11 User-Drawn-Sketch-Layer-1)
 (GlyphFn Object-12 User-Drawn-Sketch-Layer-1)
 (GlyphFn Object-21 User-Drawn-Sketch-Layer-1)
 (GlyphFn Object-22 User-Drawn-Sketch-Layer-1)

 (GlyphFn Object-9 User-Drawn-Sketch-Layer-1)))

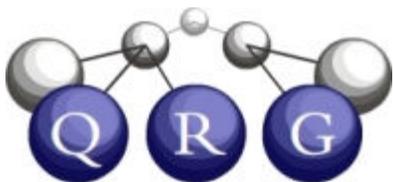
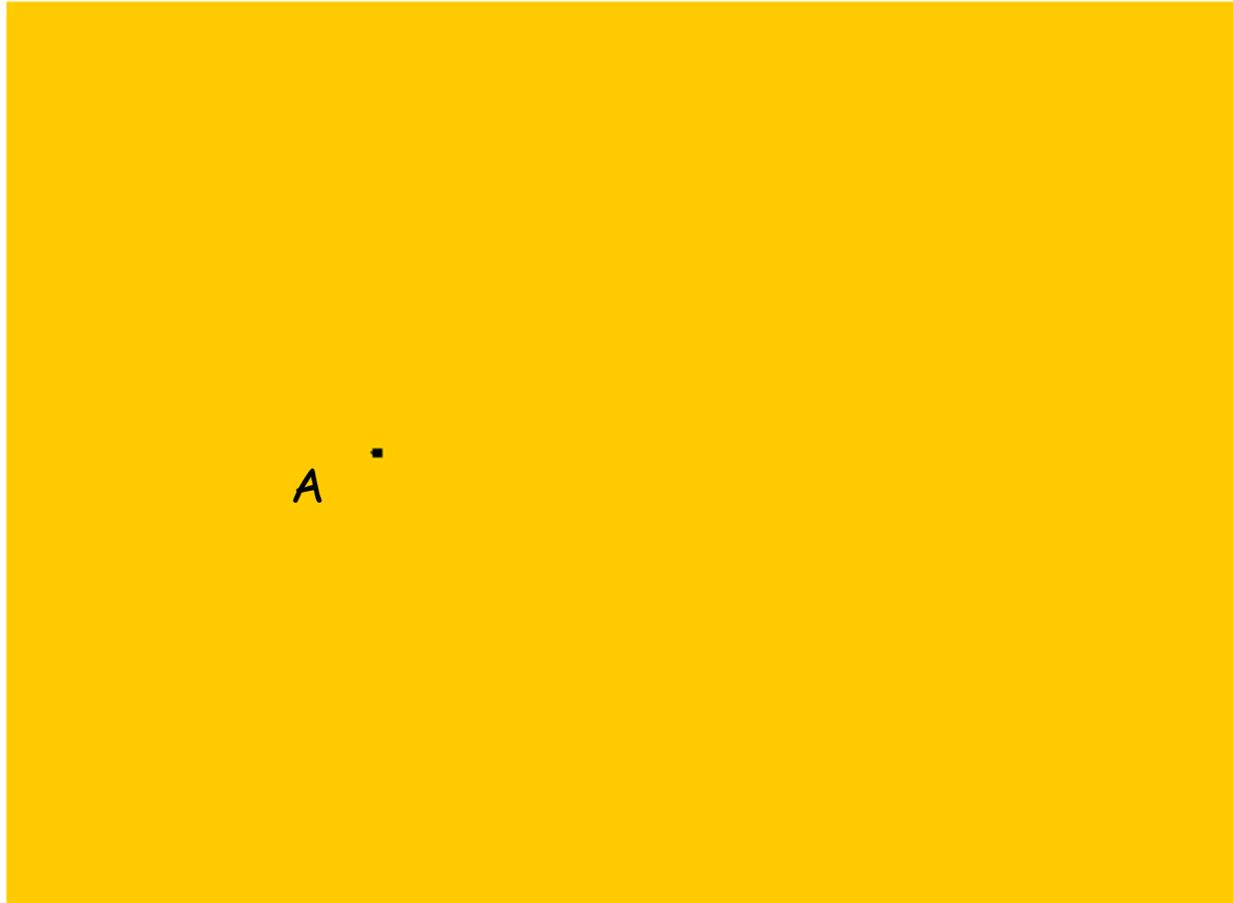


Computing Glyph Groups

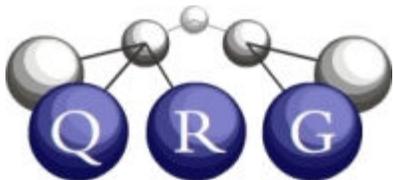
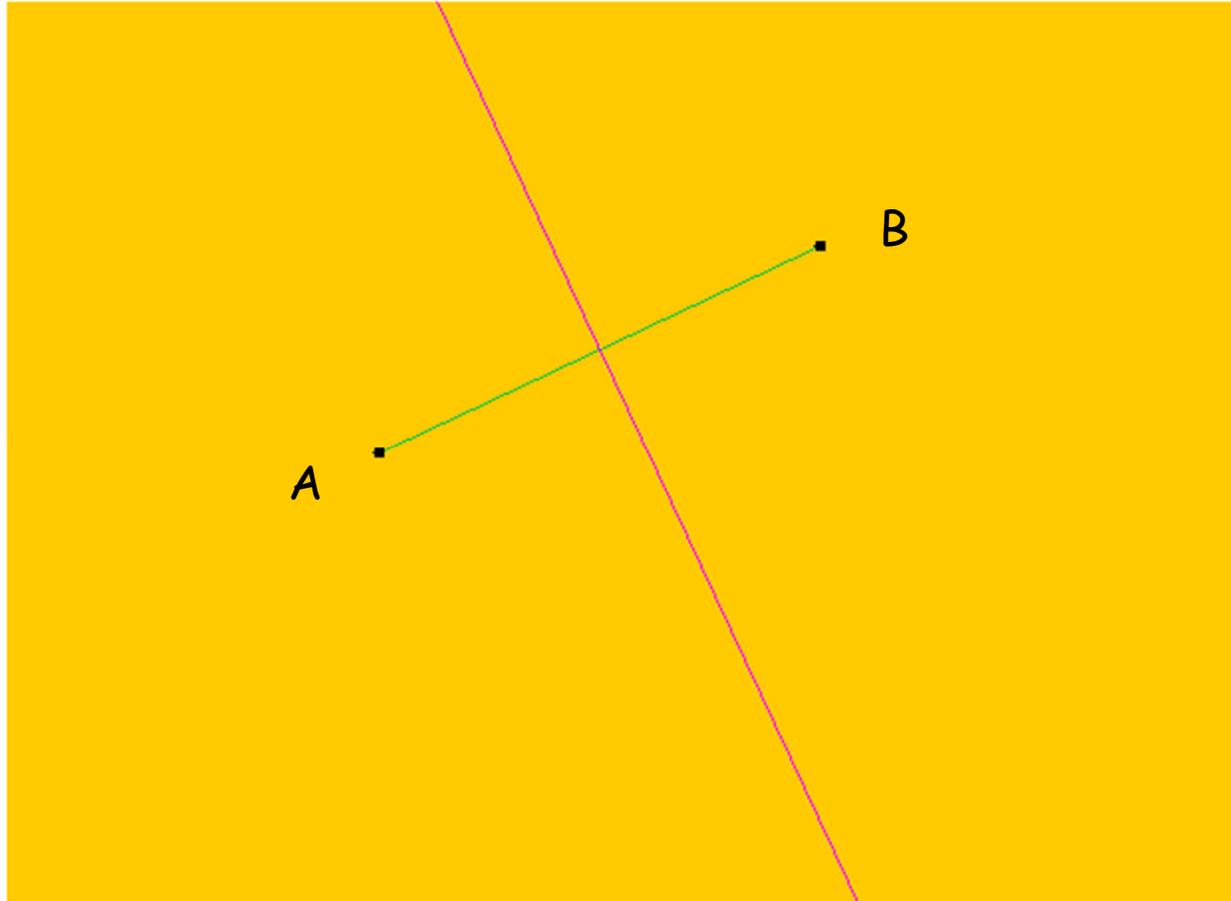
- Connection graph: Nodes = glyphs, Edges between all pairs that are EC or PO
 - Connected Glyph Groups = connected subsets of connection graph
- Containment graph: Nodes = glyphs, Edges between all pairs that are TPPi or NTPPi.
 - Contained glyph groups = All glyphs with more than one glyph inside of them, only counting directly inside glyphs
- Incrementally maintained as sketch updated



Voronoi Diagrams: A tutorial



Voronoi Diagrams: A tutorial

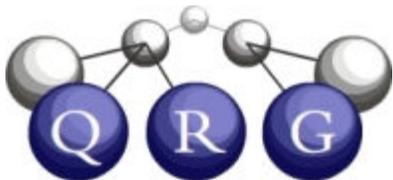
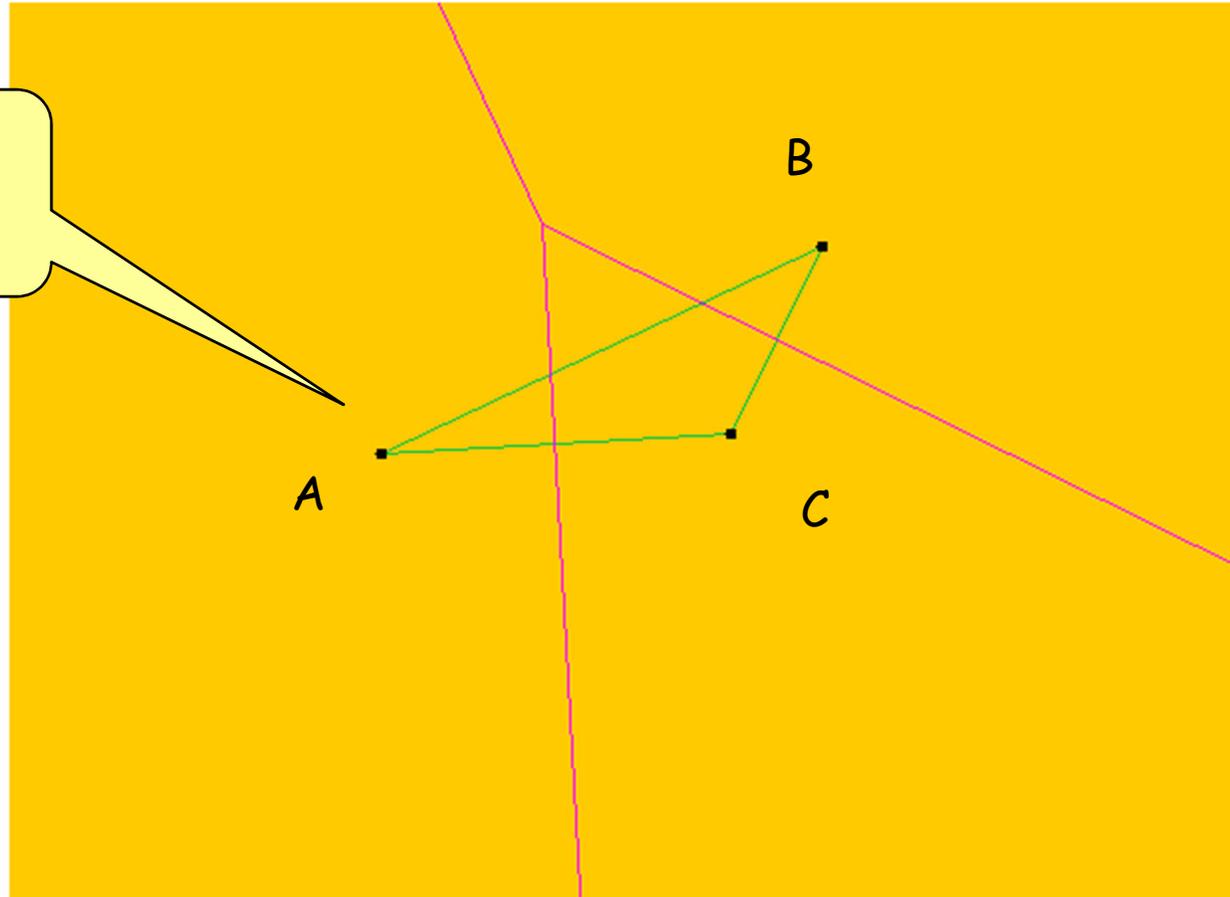


Red = cell boundary in Voronoi diagram
Green = arc in Delaunay triangulation
Voronoi diagrams and Delaunay triangulations are duals



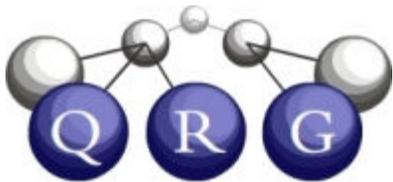
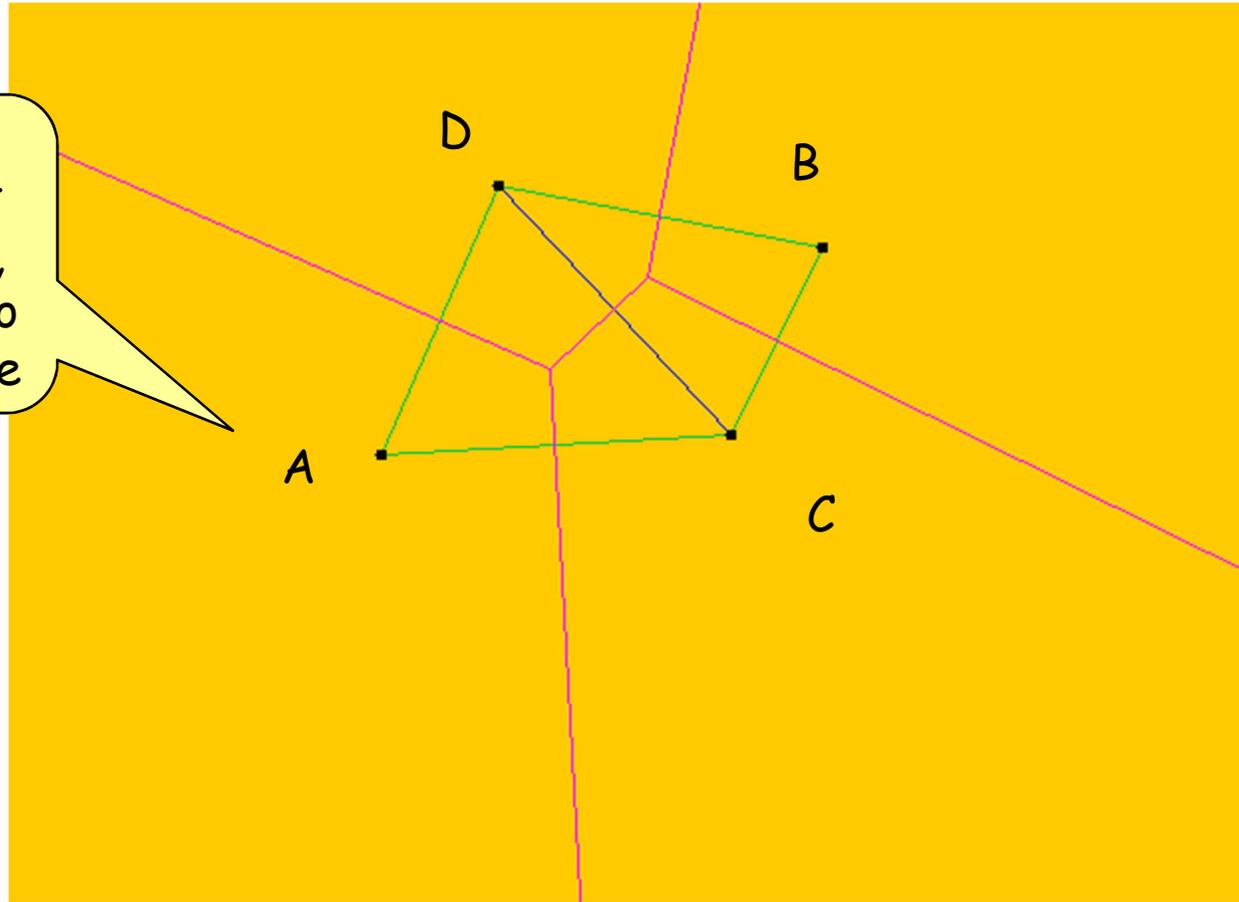
Voronoi Diagrams: A tutorial

A is adjacent to B & C



Voronoi Diagrams: A tutorial

A is adjacent to D & C, but not to B anymore

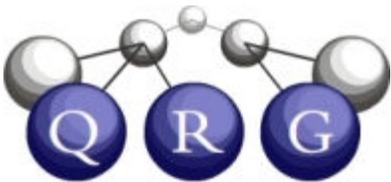
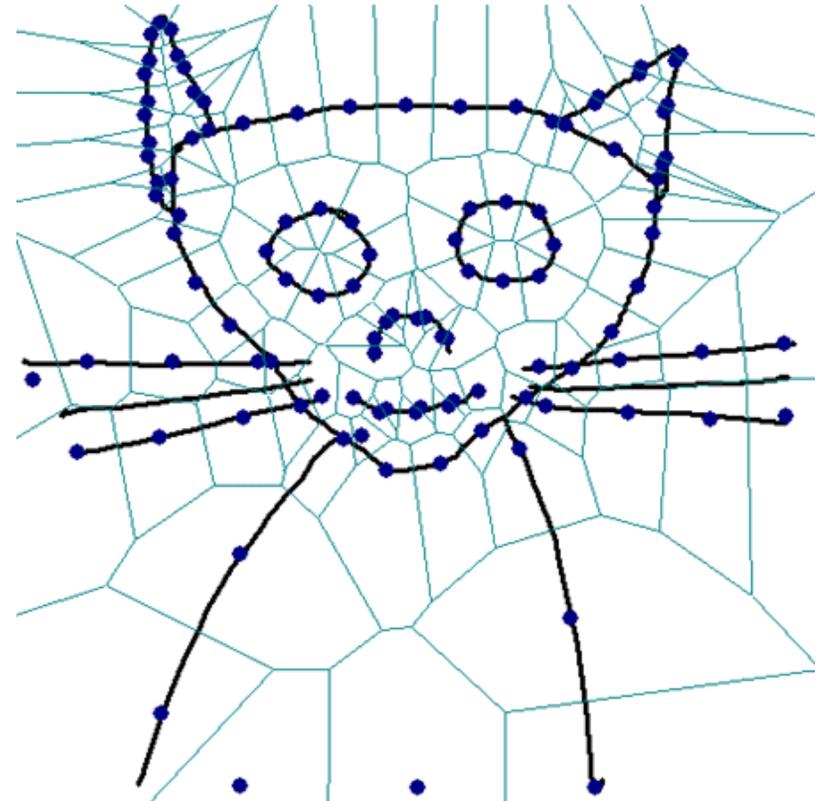


Edwards & Moulin (1998) argue that Voronoi diagrams are useful for capturing visual adjacency



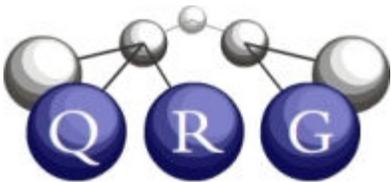
Voronoi Relationships

- Provides a notion of adjacency
- Generalizing to glyphs:
 - Use sample points along contour of glyphs to define standard Voronoi (site-level Voronoi)
 - Label edges with glyph membership
 - Define glyph-level relations in terms of site relations
 - E.g., two glyphs are **siteAdjacent**
 $\Leftrightarrow \exists$ samples on glyphs | edge-connected in site-level Delaunay triangulation
- One Voronoi diagram computed per subsketch in CogSketch



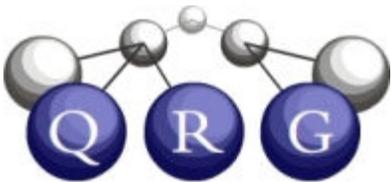
Positional Relations

- Provide qualitative position, orientation information with respect to global frame of reference
 - For glyphs, `leftOf`, `rightOf`, `above`, `below`
 - For contents, depends on genre and viewpoint
 - Physical/side: Same as glyphs
 - Geospatial/TopDown: `northOf`, `southOf`, `eastOf`, `westOf`
 - Abstract or Discrete: No implications for contents

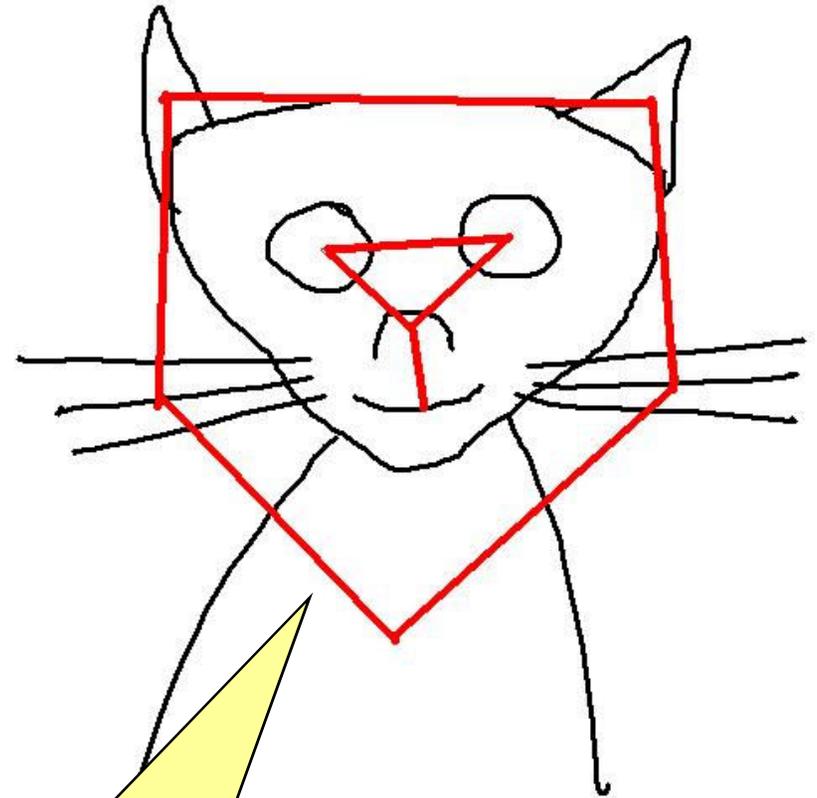
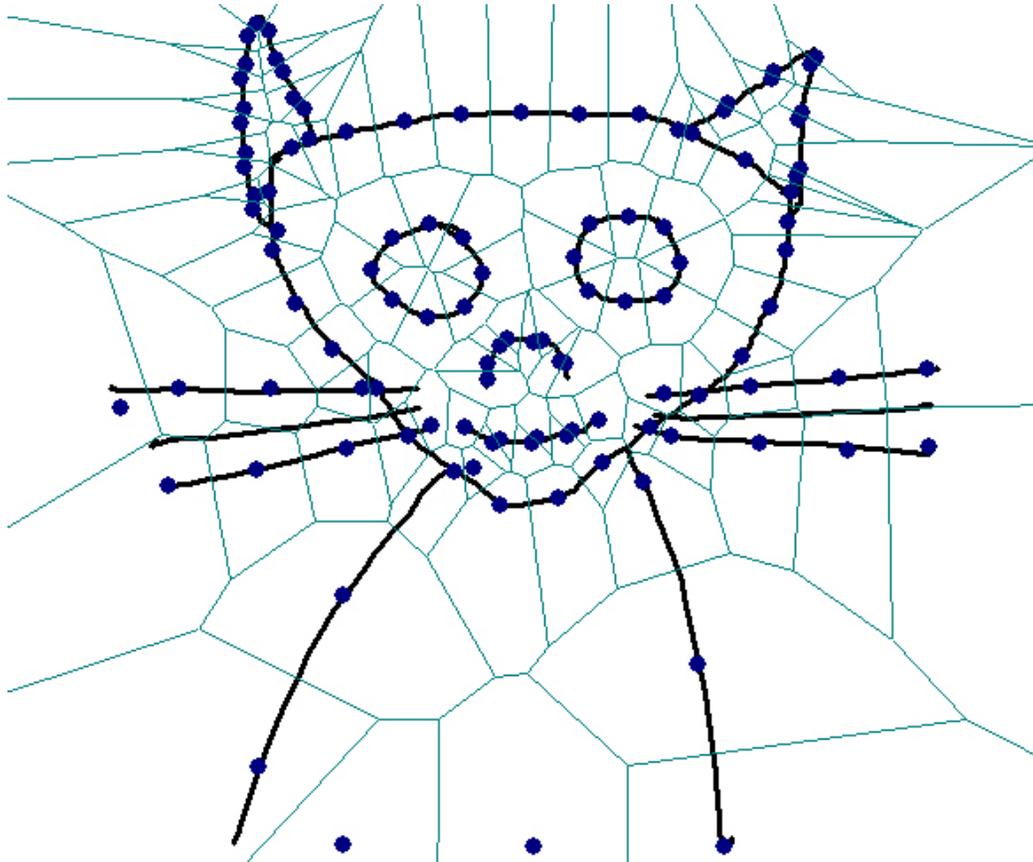


Local Relational Neighborhood Hypothesis

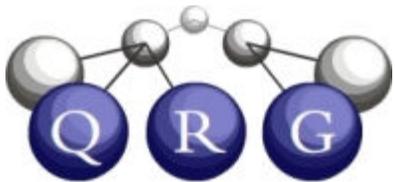
- When to compute positional relations? Between every pair of glyphs on a layer, like RCC8?
 - Bad idea! Loses locality
- Idea: Network of positional relations should provide “framing effect” in visual structure
- Necessary condition: Glyphs must be `siteAdjacent` on Voronoi diagram
 - Can also be computed on demand
- Hypothesis: This local neighborhood structure corresponds to default encoding method in human sketch perception



Voronoi adjacency guides positional relation finding



Positional relations only
created between site-
adjacent glyphs



Positional Relations help frame visual structure

Base: A | Mapping #26 (score = 0.419)

A		B
Triangle	0.232	Triangle
Circle	0.040	Circle
Meta-Layer	0.036	Meta-Layer
A	0.036	B
Glyph of Triangle.	0.005	Glyph of Triangle.
Glyph of Circle.	0.005	Glyph of Circle.
Glyph of A.	0.005	Glyph of B.
Square	0.004	Square

Target: B

Shows a mapping interface for Mapping #26 with a score of 0.419. The Base (A) contains a Triangle, Circle, and Square. The Target (B) contains a Circle, Triangle, and Square. The mapping table shows the highest score for the Triangle in the Base mapping to the Triangle in the Target.

Corresponds to what people choose in fast response-time task

Analogy Results | Mapping #32 (score = 1.024)

A		B
Triangle	0.624	Circle
Circle	0.100	Triangle
Square	0.072	Square
Meta-Layer	0.036	Meta-Layer
A	0.036	B
Glyph of Triangle.	0.013	Glyph of Circle.
Glyph of Circle.	0.013	Glyph of Triangle.
Glyph of Square.	0.009	Glyph of Square.

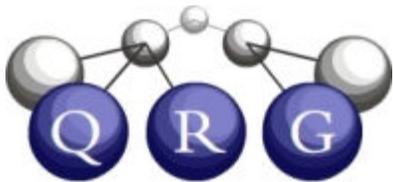
Target: B

Shows an analogy results interface for Mapping #32 with a score of 1.024. The Base (A) contains a Triangle, Circle, and Square. The Target (B) contains a Circle, Triangle, and Square. The mapping table shows the highest score for the Triangle in the Base mapping to the Circle in the Target.

Corresponds to what people choose when given more time

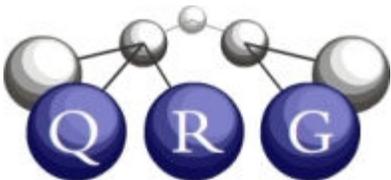


Perceptual Sketchpad



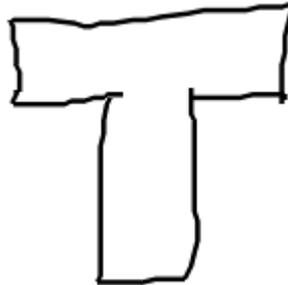
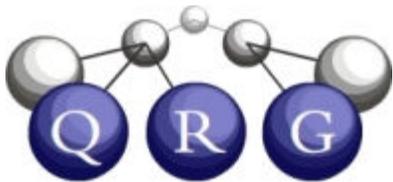
Perceptual Sketchpad Motivation

- Facility for experimenting with expressive representation of shapes
 - Decomposing glyphs
 - Within-glyph relationships also important
 - e.g., symmetry
 - Modeling mental rotation
- Still experimental, hence separate subsystem
 - Not all CogSketch users need it
 - Is now being integrated into educational worksheets



Understanding Form

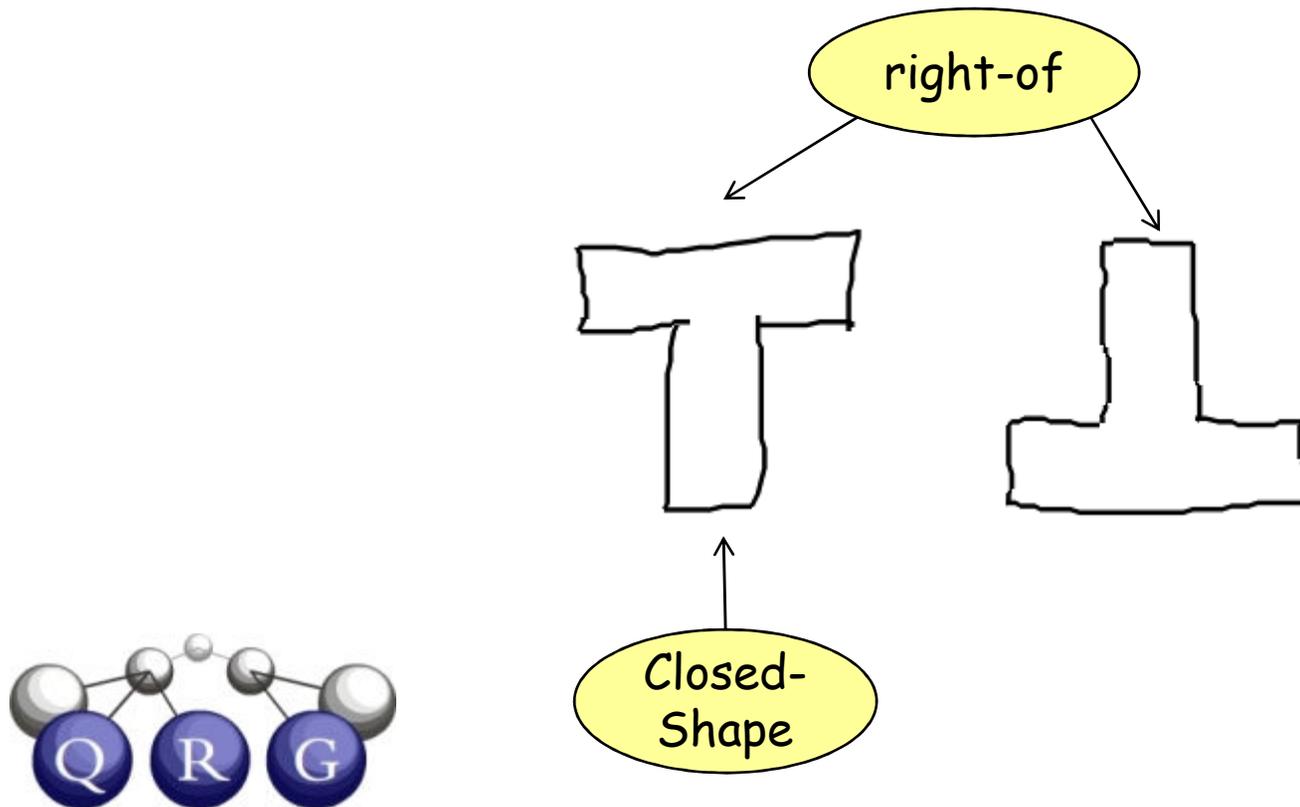
- Focus is on understanding the form of glyphs
 - *Don't* recognize a glyph
 - *Do* recognize that two glyphs are the same shape
 - Identify transformations between two glyphs' shapes
 - Scaling
 - Rotation
 - Reflection



Three Levels of Representational Focus

1) Object-Level Representation

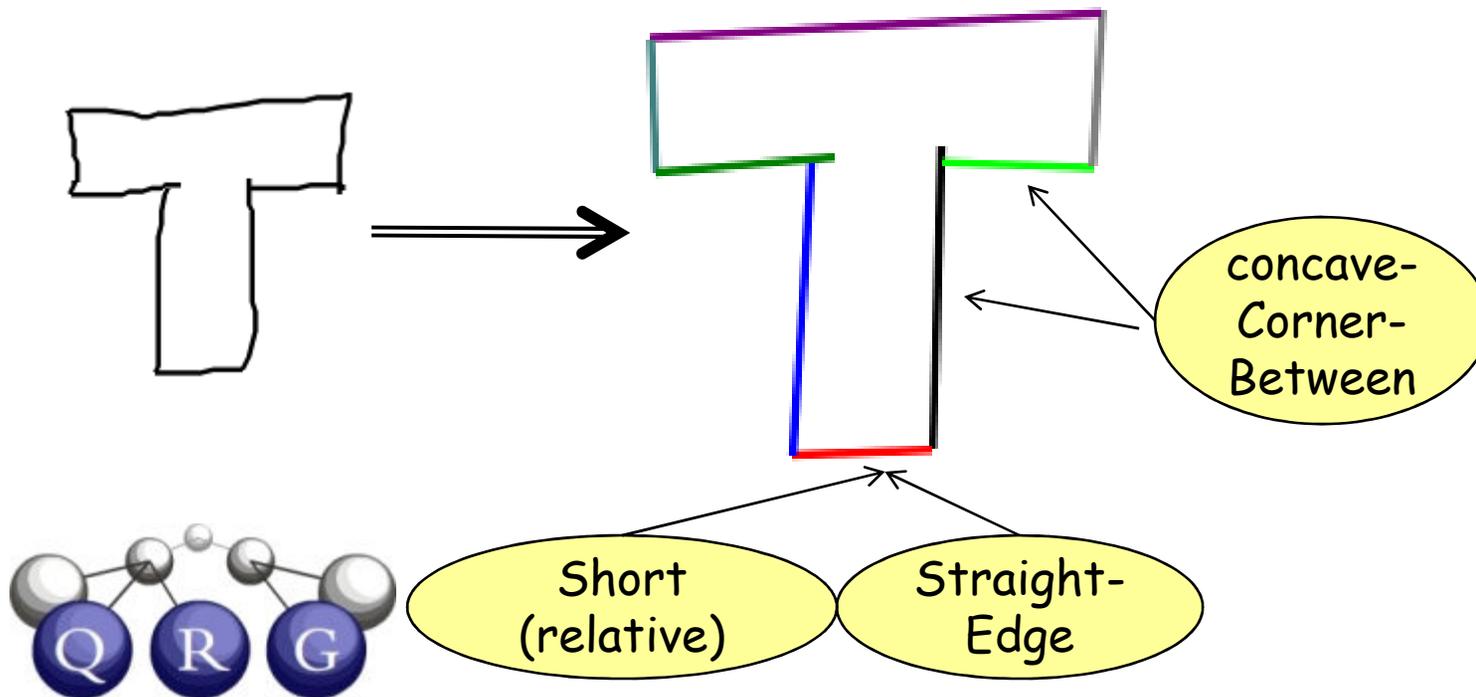
- Default CogSketch representation level
- Glyphs are the entities
- Represent attributes of, relations between glyphs



Three Levels of Representational Focus

2) Edge-Level Representation

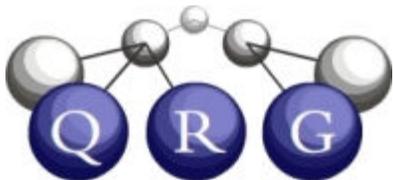
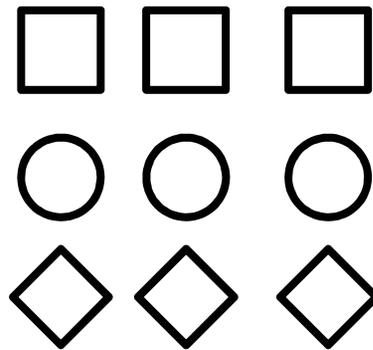
- Glyph is automatically segmented into edges
- Edges are the entities
- Represent attributes of, relations between edges within a glyph



Three Levels of Representational Focus

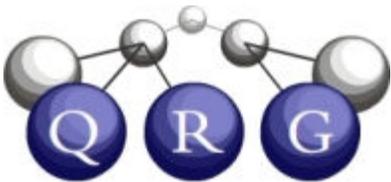
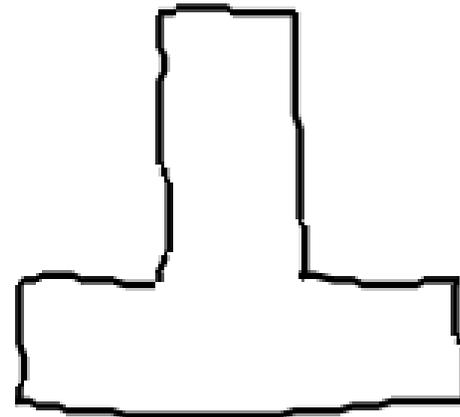
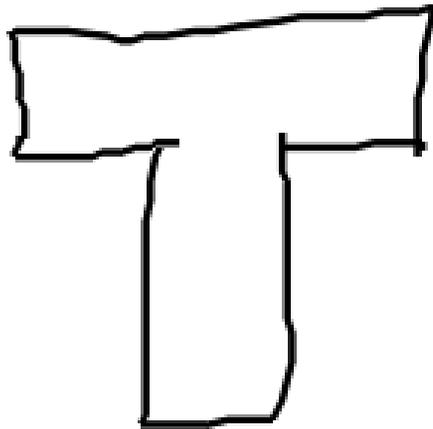
3) Group-Level Representation

- Glyphs grouped together based on similarity



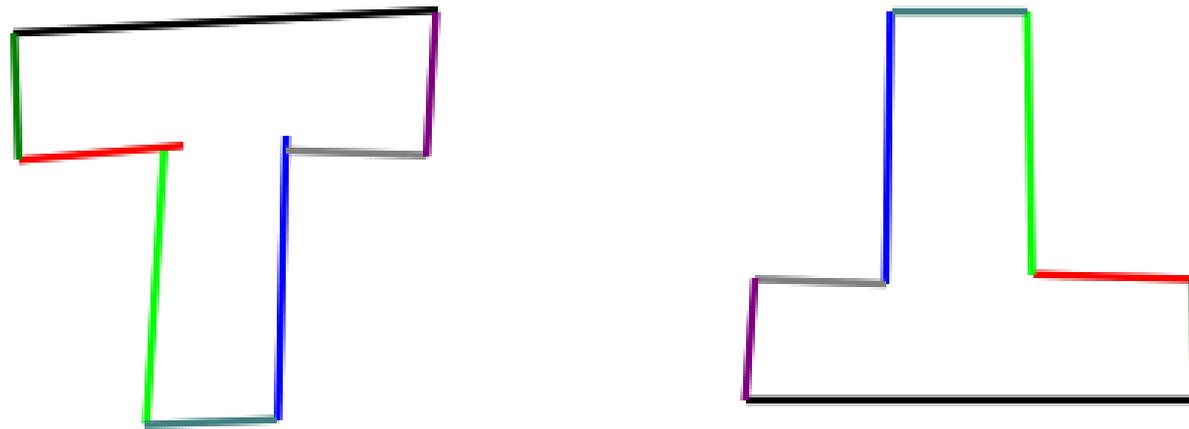
Shape Relations

- 1) Compare two glyph's edge representations to find corresponding edges
- 2) Compare orientations of corresponding edges to identify rotations or reflections

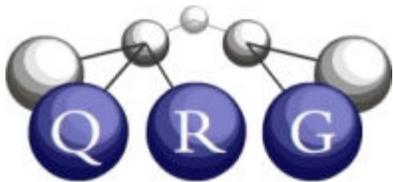


Shape Relations

- 1) Compare two glyph's edge representations to find corresponding edges
- 2) Compare orientations of corresponding edges to identify rotations or reflections

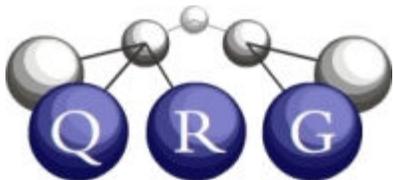
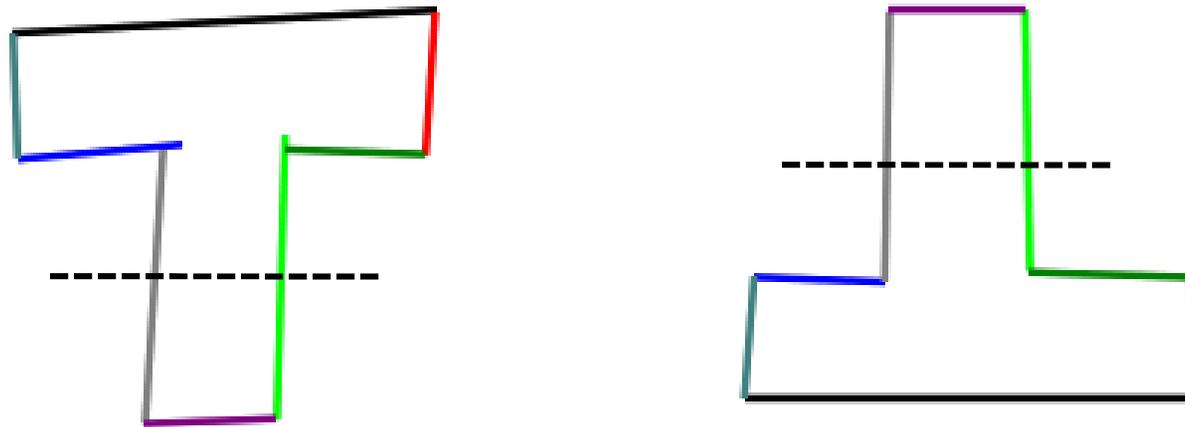


Rotation: 180 degrees



Shape Relations

- 1) Compare two glyph's edge representations to find corresponding edges
- 2) Compare orientations of corresponding edges to identify rotations or reflections



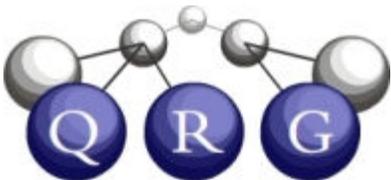
Reflection: X Axis



Sampling of Spatial Vocabulary

Objects

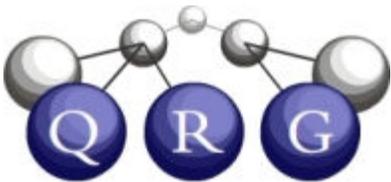
- Relations
 - Relative position
 - Topology
 - Intersect, Overlap, Contains
 - Alignment
 - Centered-On
 - Shape Relations
 - Rotation
 - Reflection
- Attributes
 - Fill/Edge color
 - Closed vs. Open
 - Relative size
 - Symmetry



Sampling of Spatial Vocabulary

Edges

- Relations
 - Corners
 - Concave/Convex
 - Relative orientation
 - Parallel/Perpendicular
- Attributes
 - Straight/Curved/
Elliptical
 - Relative length

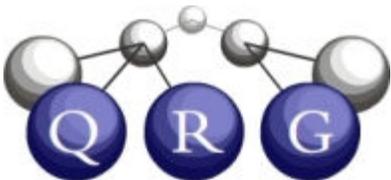


Using the Perceptual Sketchpad

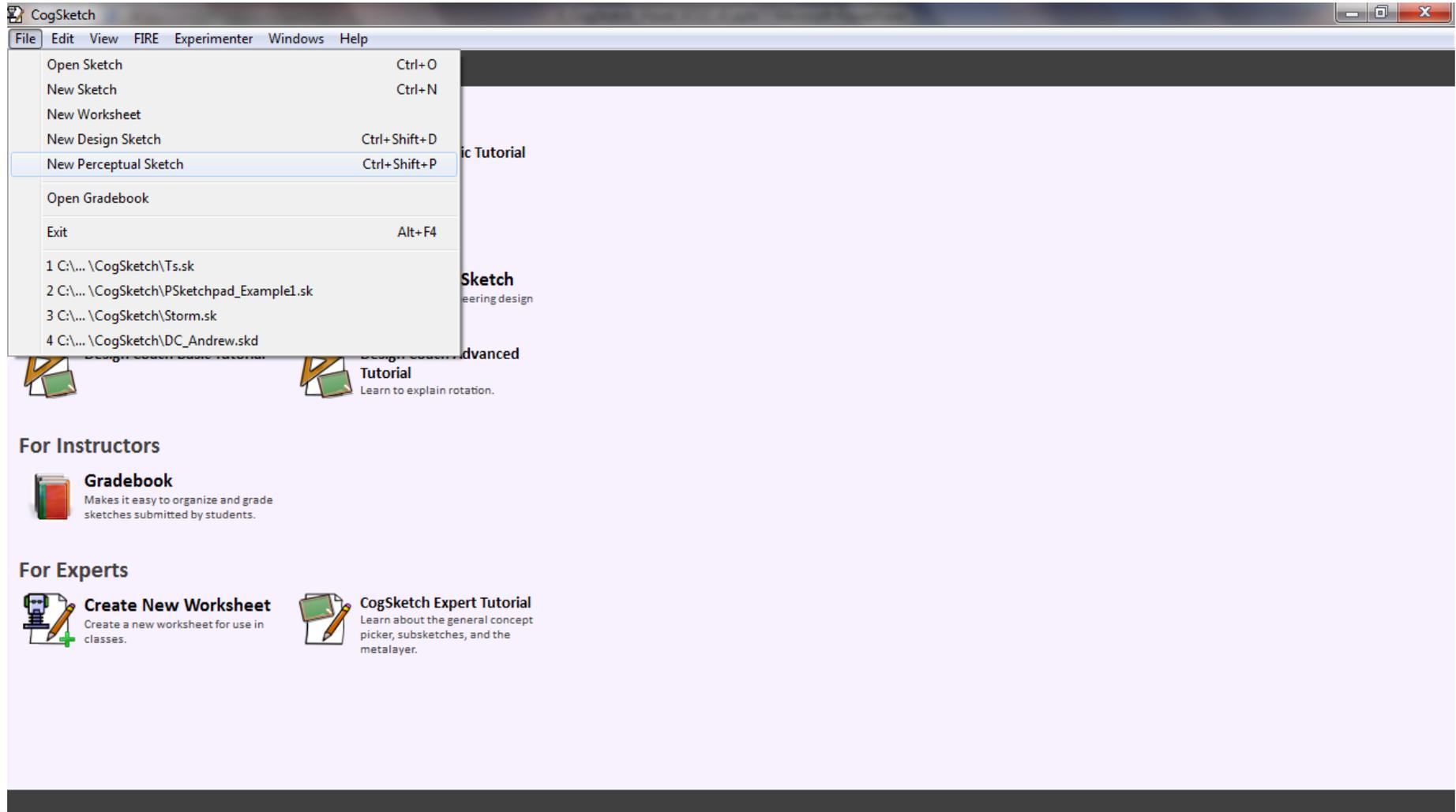
- CogSketch comes with a Perceptual Sketchpad demo
 - Choose “New Perceptual Sketchpad” from the File Menu

OR

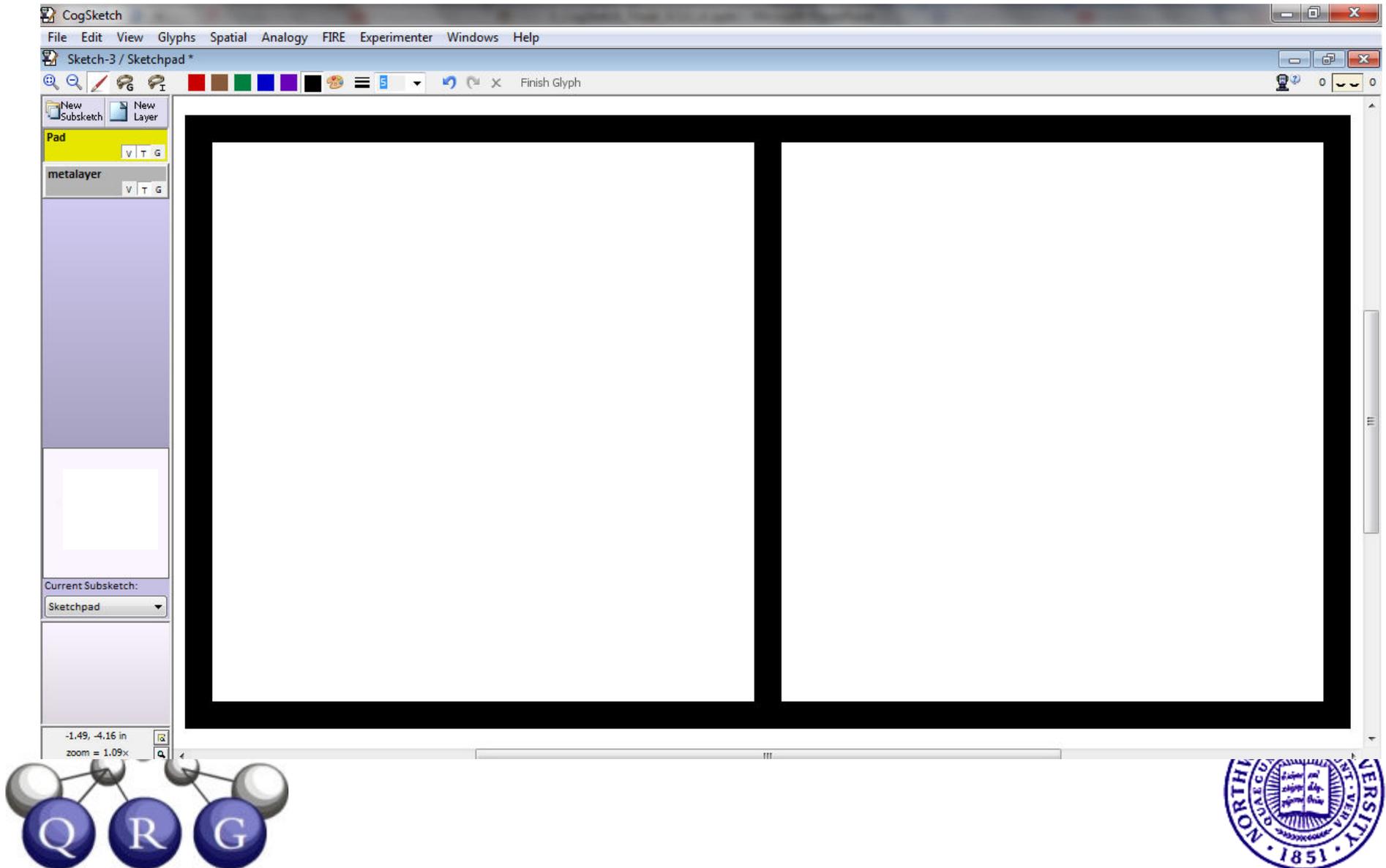
- Open one of the examples from the sketches directory
 - PSketchpad_Example1
 - PSketchpad_Example2



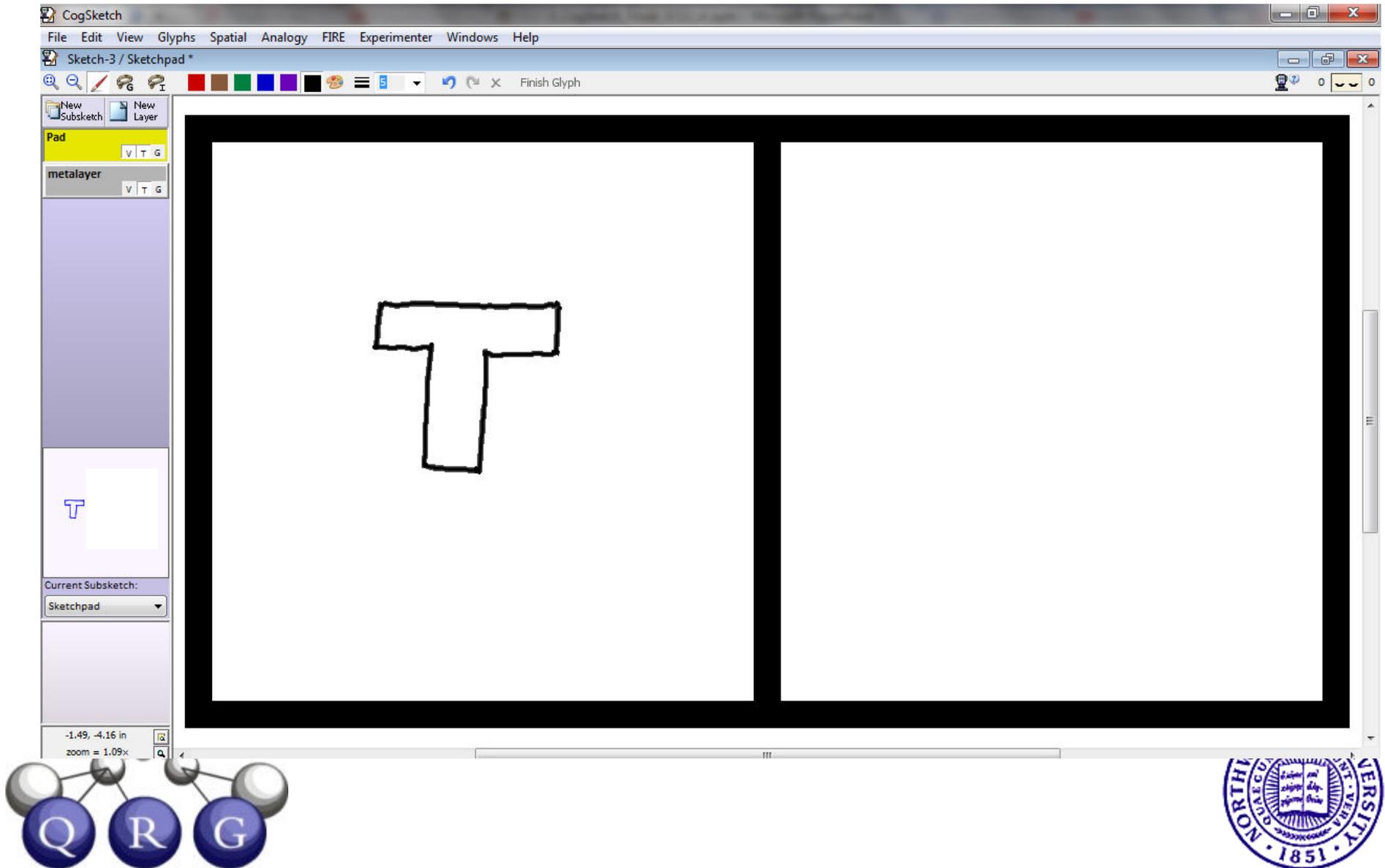
Using the Perceptual Sketchpad



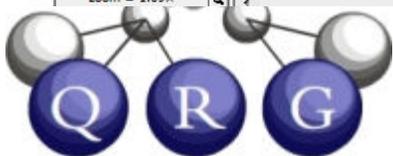
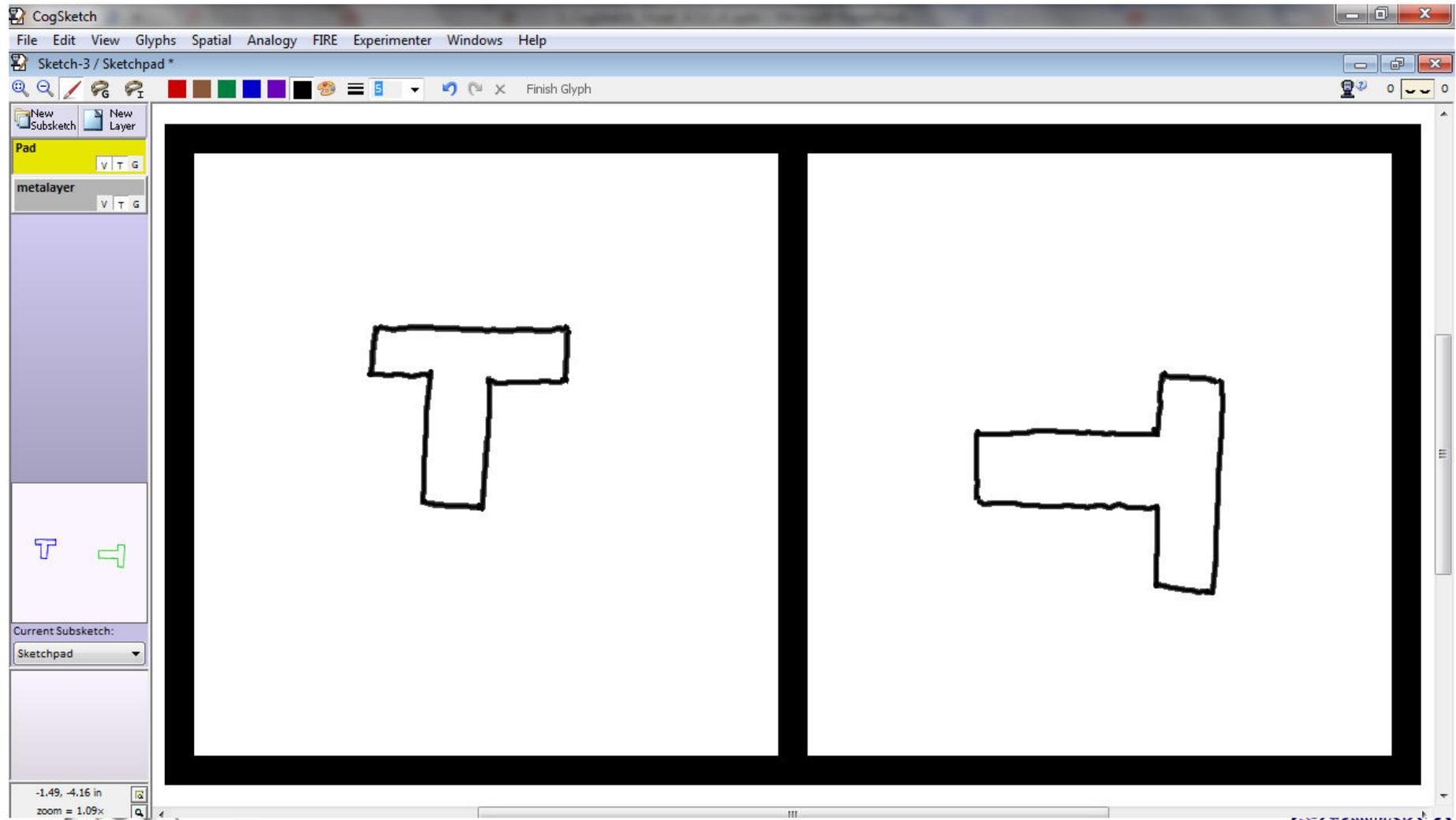
Using the Perceptual Sketchpad



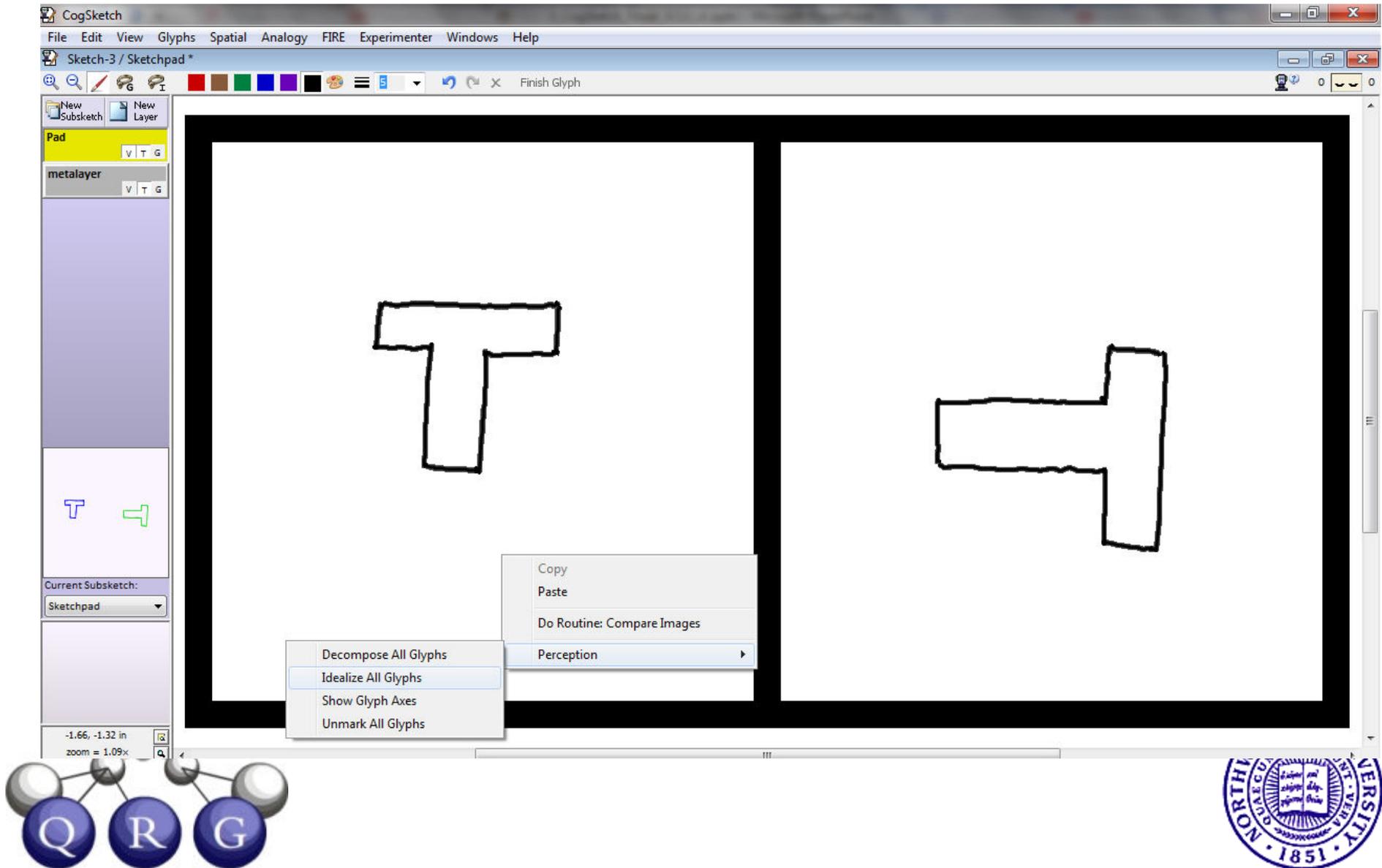
Using the Perceptual Sketchpad



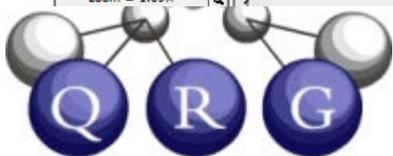
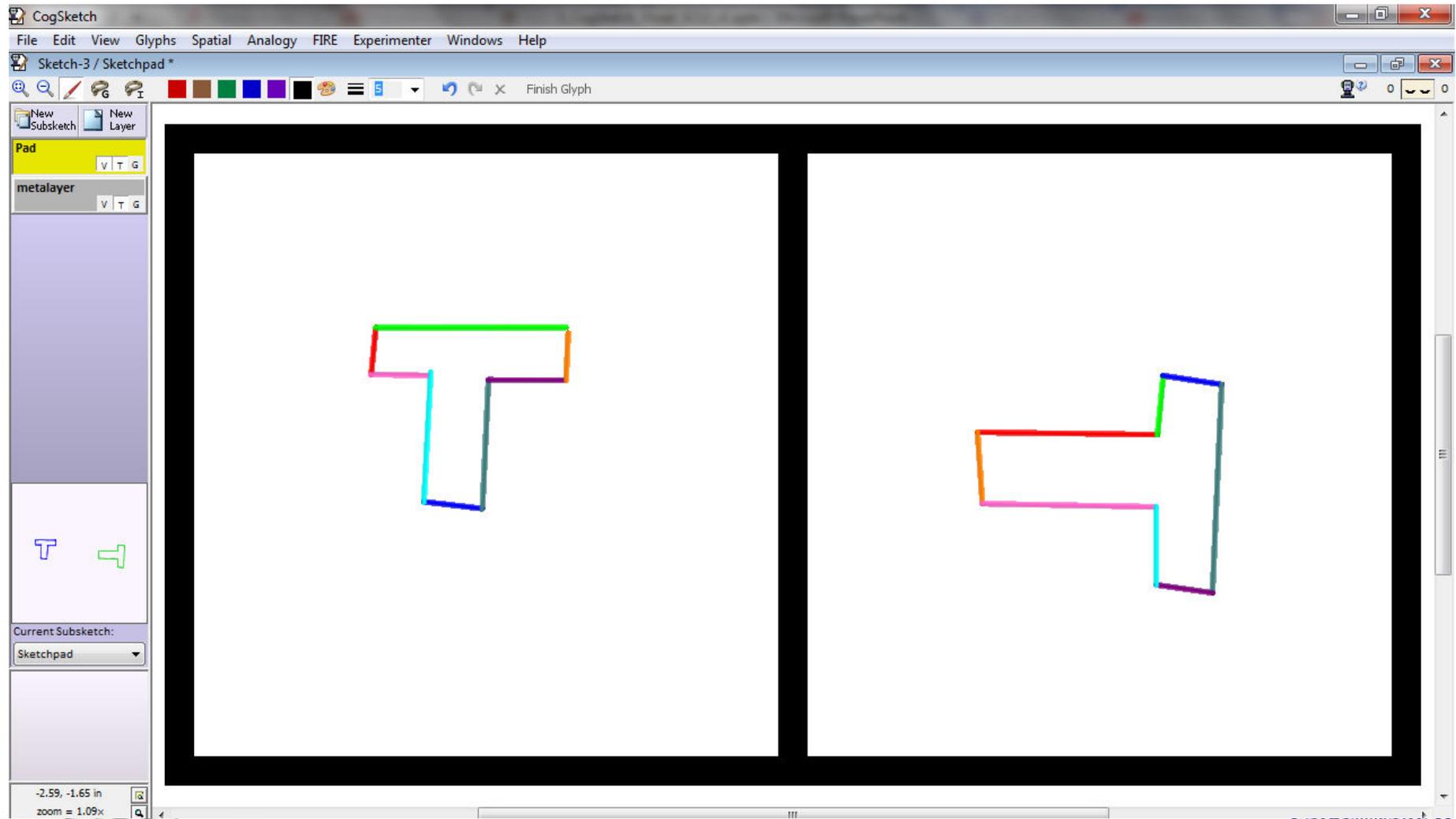
Using the Perceptual Sketchpad



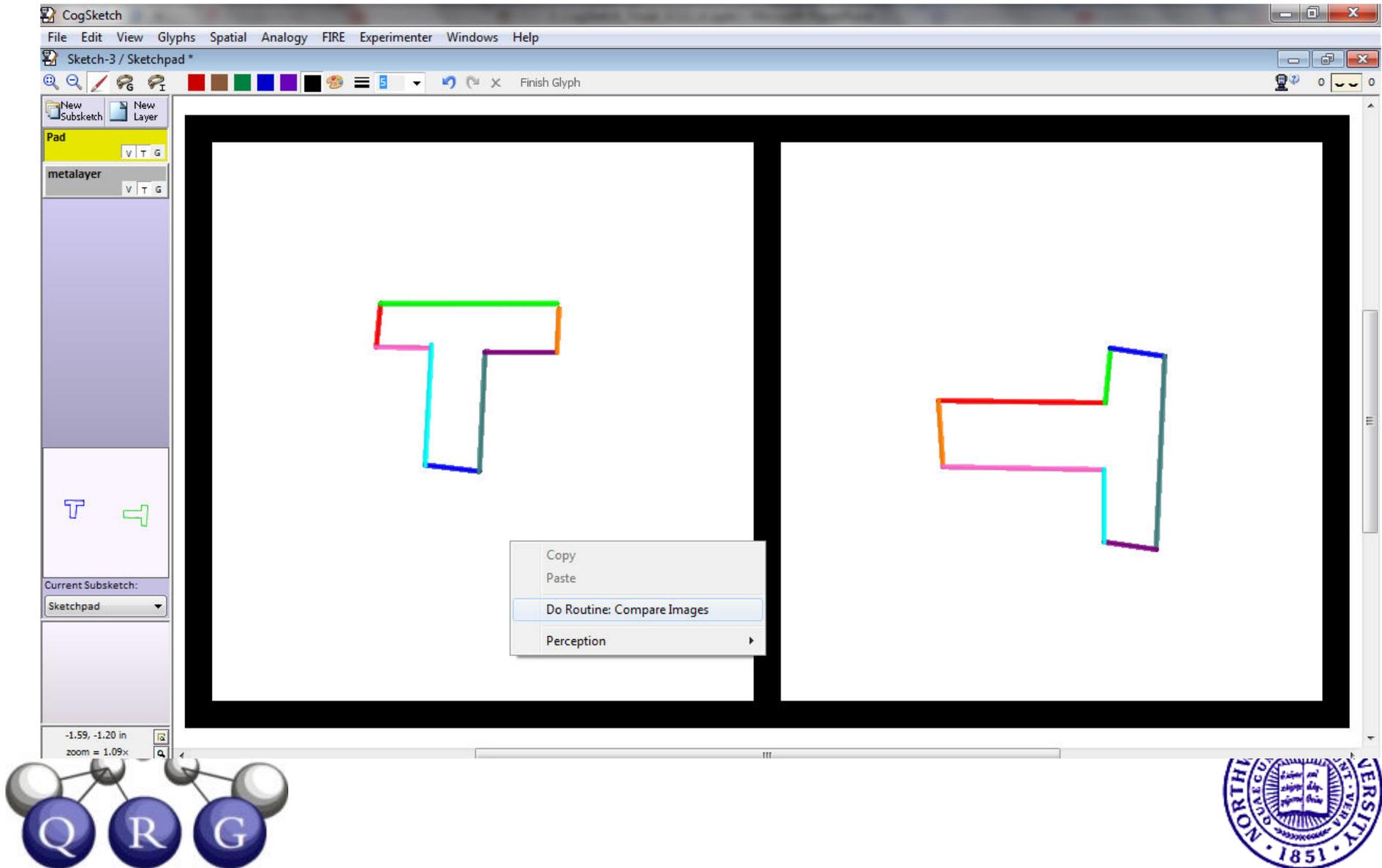
Using the Perceptual Sketchpad



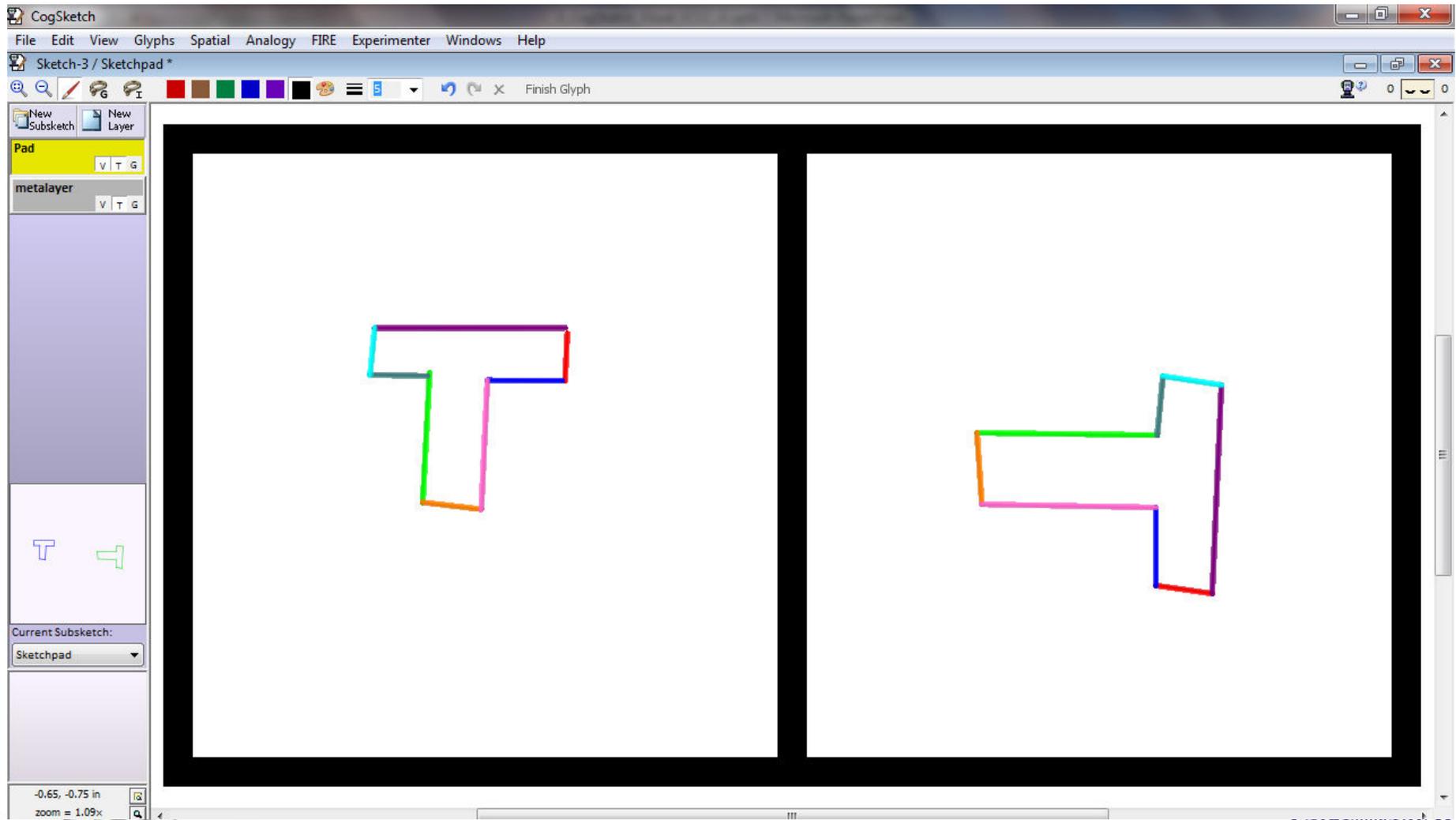
Using the Perceptual Sketchpad



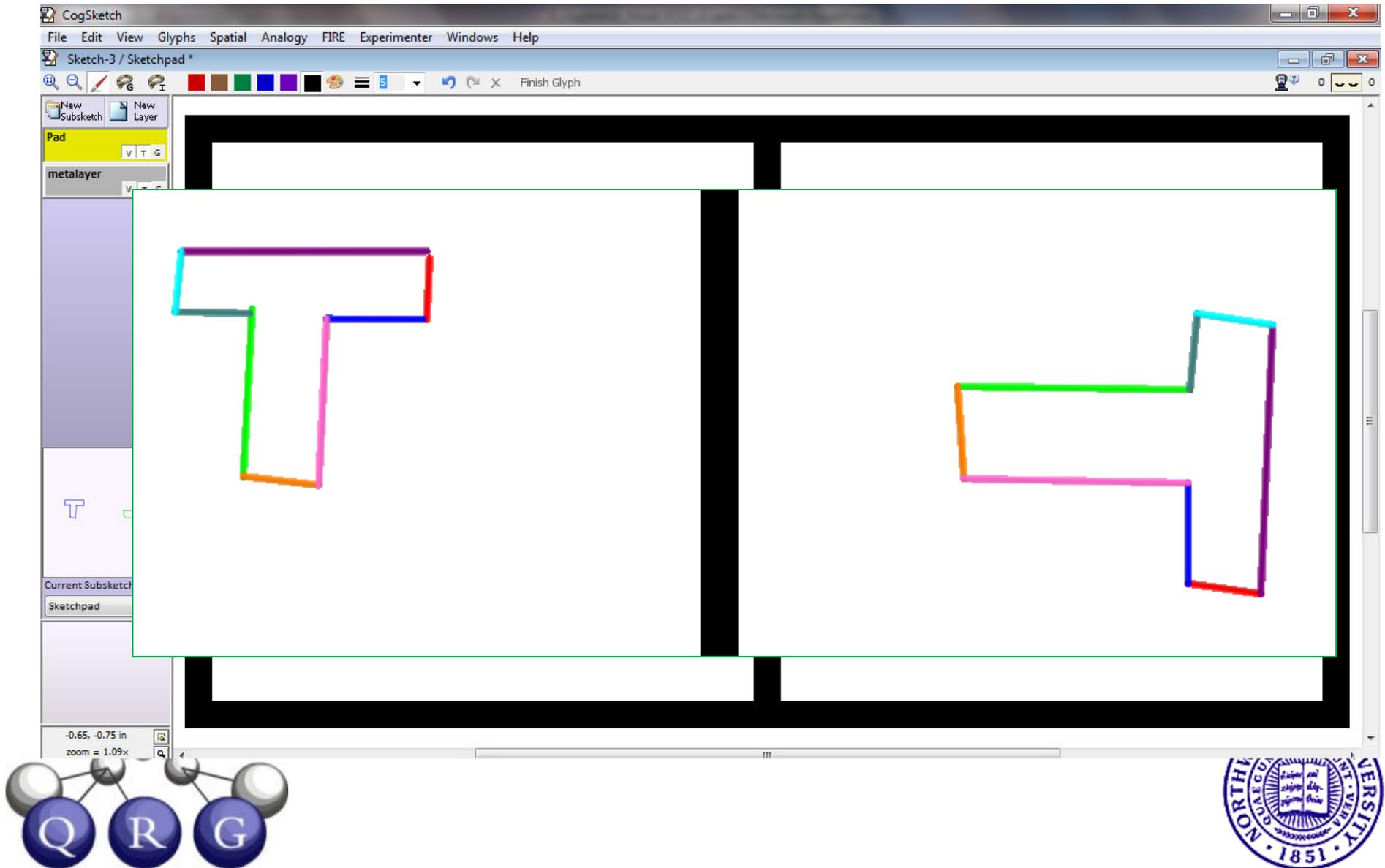
Using the Perceptual Sketchpad



Using the Perceptual Sketchpad



Using the Perceptual Sketchpad



Using the Perceptual Sketchpad

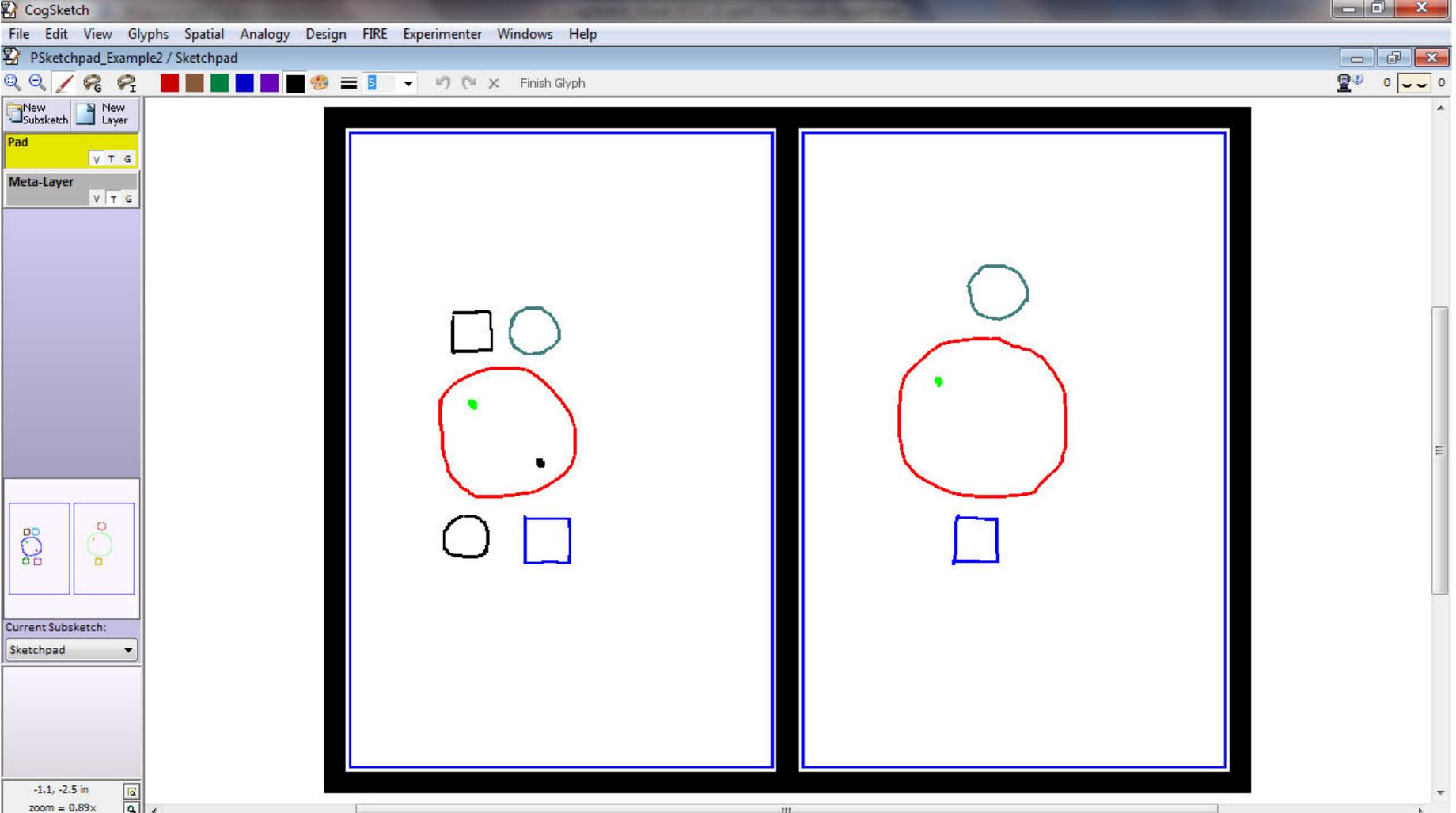
The screenshot displays the CogSketch software interface. The main window is titled "PSketchpad_Example2 / Sketchpad". The interface includes a menu bar (File, Edit, View, Glyphs, Spatial, Analogy, Design, FIRE, Experimenter, Windows, Help), a toolbar with various drawing tools, and a sidebar on the left. The sidebar contains controls for "New Subsketch", "New Layer", "Pad", "Meta-Layer", and "Current Subsketch". The main drawing area is divided into two panels. The left panel shows a sketch of a face with two small shapes (a square and a circle) above and below it. The right panel shows a sketch of a face with a small circle above and a small square below it. The interface also shows a status bar at the bottom left with coordinates and zoom level.

Q R G

PSketchpad_Example2

The seal of North Carolina State University, established in 1851, is located in the bottom right corner of the image.

Using the Perceptual Sketchpad



The image shows a screenshot of the CogSketch software interface. The window title is "CogSketch" and the menu bar includes "File", "Edit", "View", "Glyphs", "Spatial", "Analogy", "Design", "FIRE", "Experimenter", "Windows", and "Help". The main workspace is divided into two vertical panels, each enclosed in a blue border. The left panel contains several geometric shapes: a black square, a light blue circle, a large red circle with a small green square and a black dot inside, a black circle, and a blue square. The right panel contains a light blue circle at the top, a large red circle with a small green square inside, and a blue square at the bottom. On the left side of the interface, there is a control panel with "New Subsketch" and "New Layer" buttons, "Pad" and "Meta-Layer" sections with "V", "T", and "G" buttons, and a "Current Subsketch:" dropdown menu set to "Sketchpad". At the bottom left, there are three blue spheres with white letters "Q", "R", and "G" on them. At the bottom center, a yellow rounded rectangle contains the text "PSketchpad_Example2". At the bottom right, there is a circular blue seal of North Carolina State University, featuring the text "NORTH CAROLINA STATE UNIVERSITY" and "1851".

Using the Perceptual Sketchpad

- If there is one glyph in each entry
 - Edge representations will be used
- If there are multiple glyphs
 - Object representations will be used
- Elements will be color-coded to indicate correspondences
 - Right-click and choose “Unmark all glyphs” to remove colors

